BRNO UNIVERSITY OF TECHNOLOGY

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF MECHANICAL ENGINEERING

FAKULTA STROJNÍHO INŽENÝRSTVÍ

INSTITUTE OF SOLID MECHANICS, MECHATRONICS AND BIOMECHANICS

ÚSTAV MECHANIKY TĚLES, MECHATRONIKY A BIOMECHANIKY

DEVELOPMENT OF A UNIVERSAL CELL FOR OPTICAL PART INSPECTION IN A ROBOTIC WORKPLACE

VÝVOJ UNIVERZÁLNÍ BUŇKY PRO OPTICKOU KONTROLU VÝROBKŮ NA ROBOTICKÉM PRACOVIŠTI

MASTER'S THESIS

DIPLOMOVÁ PRÁCE

AUTHOR AUTOR PRÁCE	Bc. Radim Cahlík
SUPERVISOR VEDOUCÍ PRÁCE	Ing. Roman Adámek

BRNO 2022

Assignment Master's Thesis

Institut:	Institute of Solid Mechanics, Mechatronics and Biomechanics
Student:	**Bc. Radim Cahlík**
Degree programm:	Mechatronics
Branch:	no specialisation
Supervisor:	**Ing. Roman Adámek**
Academic year:	2021/22

As provided for by the Act No. 111/98 Coll. on higher education institutions and the BUT Study and Examination Regulations, the director of the Institute hereby assigns the following topic of Master's Thesis:

Development of a universal cell for optical part inspection in a robotic workplace

Brief Description:

With the advances in industrial automation, emphasis is also placed on the fact that individual solutions are not only single–purpose but are universal and can be used in a variety of different applications as well. A typical example might be an optical inspection of parts when we want to use one device to inspect a whole range of products and types of defects with only minimal requirements for modification. This thesis will deal with the development of a universal cell for optical part inspection in a robotic workplace.

Master's Thesis goals:

1. Design a universal cell for optical part inspection which will feature adjustable lighting, mounting of cameras and will be equipped with a clamping mechanism for a quick change of inspected parts.
2. Develop software for optical inspection using methods of artificial intelligence.
3. Integrate the cell for optical inspection into the robotic workplace.
4. Test the functionality and usability of this cell and its SW on a variety of products with different types of defects.

Recommended bibliography:

TRUCCO, Emanuele a Alessandro VERRI. Introductory techniques for 3-D computer vision. Upper Saddle River, NJ: Prentice Hall, c1998. ISBN 0132611082.

SOLEM, Jan Erik. Programming computer vision with Python. Sebastopol, CA: O'Reilly, 2012. ISBN 1449316549.

KAEHLER, Adrian a Gary R. BRADSKI. Learning OpenCV 3: computer vision in C++ with the OpenCV library. Sebastopol, CA: O'Reilly Media, [2017]. ISBN 1491937998.

CORKE, Peter I. Robotics, vision and control: fundamental algorithms in MATLAB. Berlin: Springer, 2011. Springer tracts in advanced robotics, v. 73. ISBN 9783642201448.

GREPL, Robert. Kinematika a dynamika mechatronických systémů. Brno: Akademické nakladatelství CERM, 2007. ISBN 978-80-214-3530-8.

Deadline for submission Master's Thesis is given by the Schedule of the Academic year 2021/22

In Brno,

L. S.

prof. Ing. Jindřich Petruška, CSc. Director of the Institute	doc. Ing. Jaroslav Katolický, Ph.D. FME dean

Abstrakt

Cílem této diplomové práce je vývoj univerzální buňky pro optickou kontrolu včetně softwaru pro detekci vad, který používá metody umělé inteligence. Dalším cílem je integrovat buňku do robotického pracoviště a otestovat ji na rozdílných dílech s různými vadami.

První část práce popisuje vývoj konceptu optické buňky a její návrh. Poté pojednává o systému pro optickou kontrolu, počínaje popisem objektivu a kamery a konče analýzou kontrolního softwaru, který pro detekci vad využívá konvoluční neuronovou síť.

Další část se zabývá vývojem robotického pracoviště včetně návrhu dvojosého robota a komunikace mezi zařízeními. Nakonec je rozebráno testování na dvou rozdílných dílech s cílem ověřit funkčnost.

Summary

The aim of this master's thesis is to develop a universal cell for optical part inspection including defects detection software that utilises artificial intelligence methods. Another goal is to integrate the cell into a robotic workplace and to test it on distinct parts with different defects.

The first part of the thesis describes the development of the optical cell concept and its design. After that, the optical inspection system is covered, starting with a description of the lens and camera, followed by the analysis of the inspection software, which uses the convolutional neural network for defect detection.

The next part deals with the development of a robotic workplace including the design of a two degrees of freedom robot and the communication between the devices. In the end, testing on two distinct parts to verify its functioning is discussed.

Klíčová slova

optická kontrola, strojové vidění, konvoluční neuronová síť, robotické pracoviště

Keywords

optical inspection, machine vision, convolutional neural network, robotic workplace

Bibliographic citation

CAHLÍK, R. *Development of a universal cell for optical part inspection in a robotic workplace.* Brno: Brno University of Technology, Faculty of Mechanical Engineering, 2022. 89 pages, Master's thesis supervisor: Ing. Roman Adámek.

Rozšířený abstrakt

Úvod

Průmyslový sektor čelil v několika uplynulých letech mnoha výzvám, včetně celosvětové pandemie, nedostatku elektronických součástek a energetické krize. Během těchto náročných časů se ukázalo, že společnosti, které dříve investovaly do moderních technologií a měly velkou část výroby automatizovanou, si s danou situací poradily snadněji. Důvodem byl fakt, že nebyly plně závislé na lidské pracovní síle a měly vyšší efektivitu.

Moderní společnost je stále více náročná z hlediska materiálních potřeb. V minulosti byl trend přesouvat výrobní závody do zemí s levnou pracovní silou. Tento přístup se nicméně ukazuje být problematický, kdykoliv vyvstane významný zdravotní nebo politický problém. Začíná být zřejmé, že domácí produkce je zásadní. Tohle nás přivádí k otázce, jak se vyrovnat s rostoucí poptávkou po produktech a snižujícím se počtem pracovníků. Odpovědí je digitalizace a automatizace, což se dá shrnout jako průmysl 4.0. Tento přechod řeší nejen spoustu problému, kterým společnosti čelí, ale také obecně zvyšuje konzistentnost dodání kvalitního zboží.

Společnost Kinalisoft s.r.o., výrobce inteligentních průmyslových zařízení, přišel s řešením nazvaným Test-it-off. Tento produkt představuje chytré robotické pracoviště, které umožňuje flexibilní editaci, manipulaci s nesetříděným materiálem a export statistik. Dá se jednoduše ovládat s využitím aplikace na běžném tabletu. Je to univerzální a uživatelsky přívětivé řešení. Myšlenka je taková, že bude několik variant, každá pro specifický typ aplikace. Vzhledem k tomu, že se jedná o nový produkt, jediná již postavená varianta je pro testování desek plošných spojů, což je možné vidět na obrázku 1.1.

Motivací pro tuto práci bylo vytvoření další varianty chytrého pracoviště zaměřené na optickou kontrolu, jelikož strojové vidění je jednou z nejužitečnějších odvětví průmyslu 4.0 a již hraje důležitou roli ve výrobním sektoru. Je spoustu aplikací strojového vidění jako třeba bezdotykové měření nebo detekce vad. Cílem bylo vytvoření optické buňky jako součásti robotického pracoviště, která bude sloužit pro testování a veletržní účely Kinalisoft s.r.o.

Závěr

Univerzální buňka pro optickou kontrolu dílů byla vyvinuta jako část projektu popsaného v této práci. Na začátku byl vymyšlen koncept vozíku, který jezdí dovnitř a ven z buňky. Tato buňka je tvořena z hliníkových desek a obsahuje vestavěné osvětlení z LED pásků, které se dá regulovat. Je umístěno za deskami z difusního plexiskla. Dále byl vytvořen poziční mechanismus, který uživateli umožnuje nastavit kameru dle konkrétní aplikace.

Vozík obsahuje rotační mechanismus, který slouží pro otáčení s dílem během pořizování snímků. Navíc je jeho součástí mechanismus pro spoušť kamery, který je tvořen kódovacím kotoučem a induktivními snímači. Dále byl vyvinut zakládací mechanismus obsahující přírubu a přípravky. Pro každý díl je konkrétní zakládací přípravek a jejich výměna je velmi rychlá a snadná.

Byla vybrána kamera a objektiv. Pro ovládání kamery byl napsán Python skript, který pracuje ruku v ruce s rotačním systémem a PLC. Poté byl vyvinut software pro detekci vad, který využívá přednaučenou konvoluční neuronovou síť. Byl vytvořen systém Python skriptů sloužící pro rychlé a účinné vytvoření datasetu, natrénování konvoluční neuronové sítě a jejího otestování.

Jako součást projektu bylo navrženo a postaveno kompletně nové pracoviště, které pracuje ve dvou módech. První slouží pro Test-it-off testování desek plošných spojů a druhé pro optickou buňku. Pracoviště zahrnuje manipulátor se dvěma stupni volnosti, který byl navrhnut a naprogramován. Nese buď vozík nebo robotický uchopovač. Navíc je součástí pracoviště Aubo i-5 kobot. Aby pracoviště fungovalo, byla vytvořena logika stavového automatu, která využívá Modbus protokol k navázání komunikace mezi PLC, PC a Aubo i-5 kobotem.

Dva odlišné díly byly vybrány k otestování funkčnosti celého produktu. První byl tlumič se dvěma typy vad a druhý byl kovový díl s jedním typem vady. Byly natrénovány konvoluční neuronové sítě s využitím systému Python skriptů a vyvinuté metodologie. Za účelem testování byly díly nechány projet pracovištěm a monitorovaly se chyby. Finální výsledky potvrdily funkčnost produktu, jelikož kontrola tlumičů měla 100% přesnost a kontrola kovových dílů 92.5% přesnost, což bylo zapříčiněno nejasnou hranicí mezi správnými a vadnými kusy.

Tento systém ještě bude muset být zaintegrován do konceptu Test-it-off a má prostor pro zlepšení, nicméně byl vytvořen dobře fungující komplexní systém, který splňuje cíle této práce.

I hereby declare that this master's thesis with the title *Development of a universal cell for optical part inspection in a robotic workplace* is my own original work developed under the supervision of Ing. Roman Adámek without the help of a third party and only using the literature listed in the bibliography.

Radim Cahlík

Brno . . . .

I would like to thank my tutor Ing. Roman Adámek for his willingness to give me advice when needed. Also, the company Kinalisoft s.r.o. deserves my gratitude for providing me with the necessary resources.

Radim Cahlík

Contents

1 Introduction

The industrial sector faced many challenges in the last few years, including the global pandemic, lack of electronic components, and the energy crisis. During these difficult times, it was proven that the companies that had invested in the modern technologies and had a high percentage of their production automated, handled such situations more easily. This was thanks to the fact that they were not fully dependant on the human workforce and had higher production efficiency.

The modern society is becoming increasingly demanding in terms of material needs. In the past, the trend was to build production sites in countries with a cheap labour force. However, this approach turns out to be problematic whenever a major health or political problem arises. It has become clear that domestic production is crucial. This brings us to the question of how to deal with the increasing production demand and the decreasing number of workers. The answer is digitisation and automation, which can be summarised as industry 4.0. Such transition not only solves many issues that companies are facing, but it also generally raises the consistency of delivering a high-quality product.

The company Kinalisoft s.r.o., a manufacturer of smart industrial machines, came up with a solution called Test-it-off. This product represents a smart robotic workplace, allowing flexible editing, manipulation with unsorted material, and statistics export. It is easily operated using an app on a conventional tablet. It is a universal and user-friendly solution. The idea is that there will be a number of different variants, each for a specific type of operation. As this is a brand-new product, the only variant built was for Printed Circuit Board (PCB) testing, which can be seen in the picture 1.1.

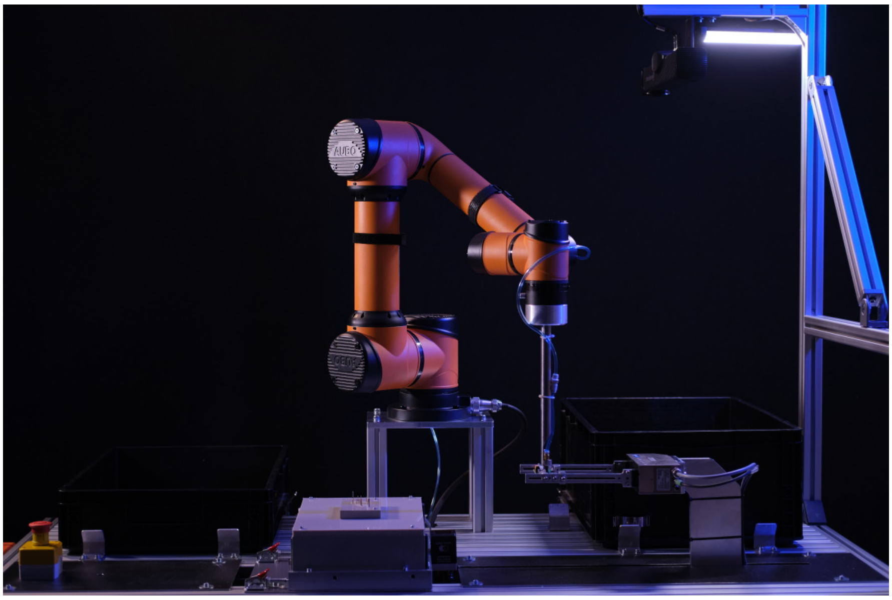


Figure 1.1: Test-it-off [1]

1 INTRODUCTION

The motivation for this thesis was to create an additional variant of the smart workplace aimed at optical inspection because machine vision is one of the most beneficial branches of industry 4.0 and already plays a crucial role in the production sector. There are many machine vision applications, such as contactless measurement or defect detection. The goal was to create an optical cell as part of the robotic workplace that would serve for testing and trade fair purposes of the Kinalisoft s.r.o.

2 Thesis objectives

The first goal was to design and build a universal cell for parts inspection. These parts were not known and could be diverse. For that reason, it was decided to make an adjustable lighting. This feature, along with the ability to set the camera into various positions, should give the user room to set up the conditions according to the application. Furthermore, there should be a mechanism that allows quick changes of the clamping systems to save time when testing various parts because each of them was supposed to have a different clamping system.

The next goal was to develop software capable of detecting defects in inspected parts using artificial intelligence (AI) and distinguishing whether the part is faulty or not. This software was to include handling communication with the camera, a script for creating a dataset, a script for training the convolutional neural network (CNN), and the final script for defect detection.

Another goal was to integrate the optical cell into the robotic workplace, which should serve as a demo unit mainly for trade fair purposes afterwards. A whole new workplace was to be built and developed as part of this goal.

The last goal was to test the functionality and usability of the cell. The testing was to be conducted on sample parts with various shapes and defects. Each part was to have its own neural network trained from a dataset obtained under certain light conditions.

3 Theoretical survey

This chapter is presenting information related to the topic of this thesis. First, the machine vision applications are discussed. This includes showing some model applications from other manufacturers. Then it deals with optical systems in machine vision. The main focus of this section is on the principle of camera and lens functioning and the advantages of each design. Lastly, this chapter shows basics of neural networks (NN) and their convolutional variant. It presents pre-trained CNNs and compares them. Furthermore, it examines software frameworks commonly used for CNNs.

3.1 Machine vision applications

What is machine vision? Definitions vary but it is basically a technology that extracts information based on an image acquired by some imaging device on an automated basis [2] and uses it for the subsequent logic. Machine vision is related to computer vision. We usually refer to machine vision when it is an application in the industrial environment, whereas we refer to computer vision when applied across the fields such as research in the replacement of human vision by computer vision. This section is focused on the machine vision because the project described in this thesis is an industrial application.

There are several types of machine vision uses. The most common ones are defects detection, completeness control, contactless measuring and robot guidance. Each type of application has its own hardware requirements which depend on customer demands, types of controlled product and other environmental conditions. These types will be discussed in the following subsections.

Machine vision can also be divided into two categories based on the software approach. The first one is using a rule-based system which, according to [3], generates output based on rules that are hard-coded by humans. It uses a series of simple IF-THEN statements. This makes the application easier for programming and the software designer has everything under control, as he creates all the logic. However, the main advantage is probably the fact that it is not data-hungry and does not require high-performance hardware. This might be the decisive factor because the client sometimes struggles with providing the sample parts or the data necessary for the second software approach, which is machine learning.

On the other hand, machine learning uses some mathematical tool, usually a neural network, which has a universal architecture and the logic is created during training. During this process, it creates its own set of rules enabling correct decision making. It is described more thoroughly in section 3.3. This approach is more modern and becomes very common. The advantages are that it can be improved in time as the network may be further trained, it also generally adapts to changes more easily and can be used in cases where it is difficult to set the rules. It is thus a tool that might find patterns and rules that are very difficult to get set using image processing and hard-coded logic.

The picture 3.1 below shows when the manufacturer of smart cameras, the company

Cognex, recommends using rule-based traditional machine vision approach and when machine learning to their clients.

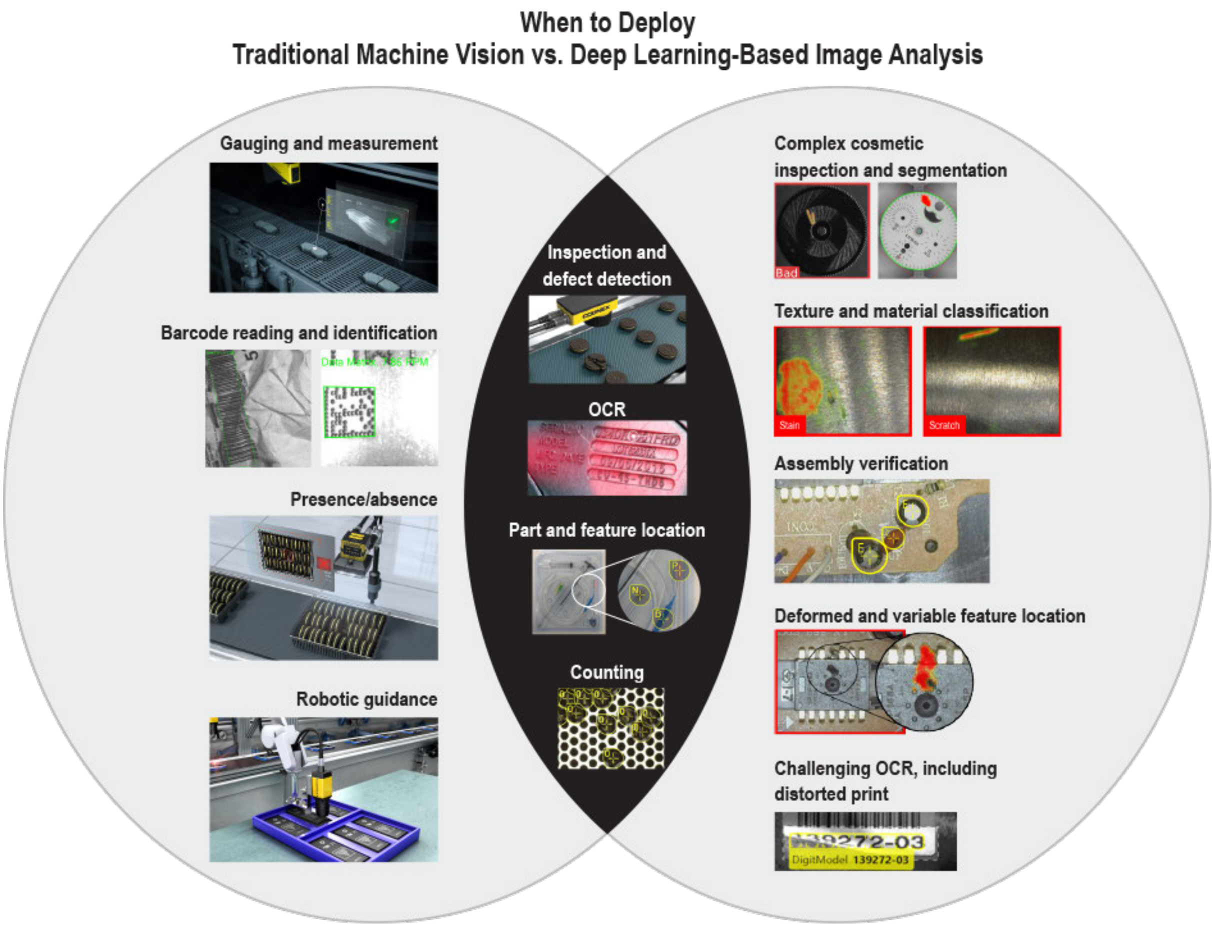

When to Deploy
Traditional Machine Vision vs. Deep Learning-Based Image Analysis
Gauging and measurement
Barcode reading and identification
Presence/absence
Robotic guidance
Inspection and defect detection
OCR
Part and feature location
Counting
Complex cosmetic inspection and segmentation
Bad
Texture and material classification
Stain
Scratch
Assembly verification
Deformed and variable feature location
Challenging OCR, including distorted print
DigitModel 139272-03

Figure 3.1: Rule-based approach vs. Machine learning [4]

3.1.1 Defects detection

The main motivation for companies to purchase devices for automated quality inspection as a replacement for the human worker is that over a longer period of time the human worker loses focus and moves the line between what is a defective part and what is not. According to [5], when a human worker sees a huge number of correct parts, the image gets imprinted on the brain and when the faulty part comes along, the worker does not see it. Due to this, the human worker lets 20% of defected parts sneak through on average, whereas the quality of work provided by a machine vision system does not deteriorate in time. This is why big companies like Heineken, where thousands of bottles come through the inspection every hour, has a high percentage of automated quality control. According to [6], another disadvantage of the human worker is that they can process only 10–12 images per second and can stay focused only on one thing, whereas a machine vision system might theoretically process an unlimited number of images as it is only limited by the image rate of the imaging system and the computing performance.

A modern solution for the defects detection using neural networks was created by the company OptiSolutions s.r.o. in cooperation with the company stoba Holding GmbH &

Co KG. and is called AI Inspector One [7]. This complex device can be seen in picture 3.2. It features a Beckhoff XPlanar system [8] that uses levitating magnetic boards carrying the parts. It enables stacking up the boards in a line to minimise the wasted time. The workplace contains multiple stations. One is used for measuring with telecentric lenses and the others are used for quality inspection and utilise indirect reflected lights along with multiple cameras set up to see the part at various angles. Neural networks are used to detect defects. This solution won the gold medal at the 2021 MSV in Brno [9].

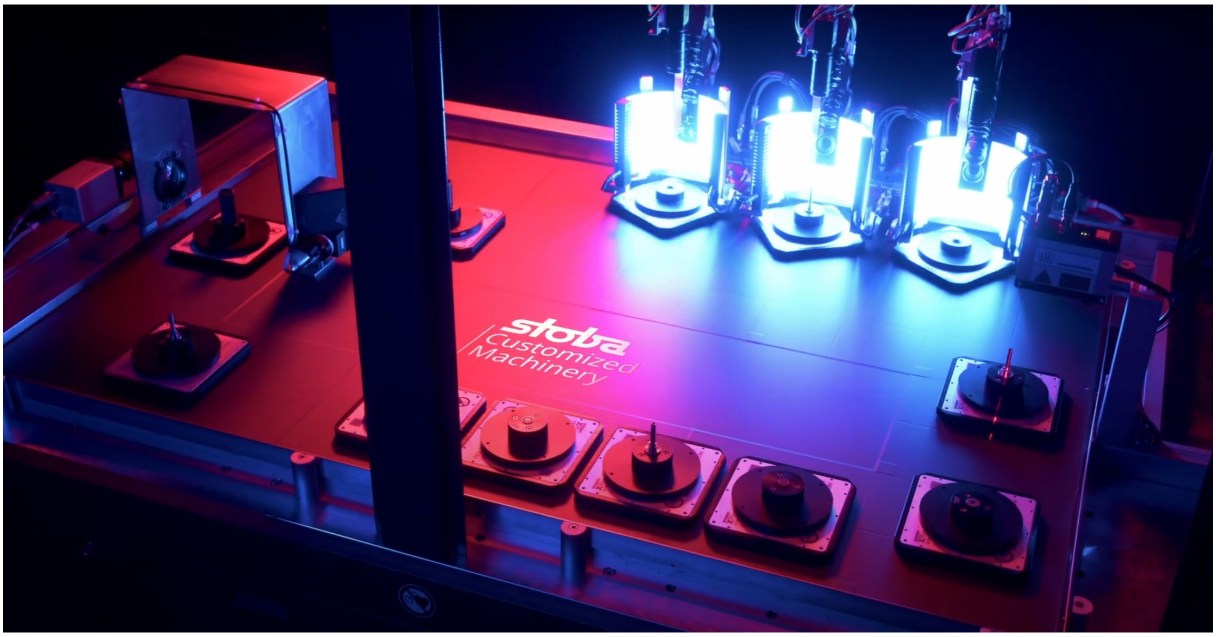


Figure 3.2: AI Inspector One [7]

Another high-end solution for defects detection was developed by the company Carl Zeiss AG. It is a German company with a long tradition and one of the world leaders in opto-electronics. Their product is called Surfmax [10] and allows universal uses in the area of surface defects detection. One of the main advantages of this machine is that it can detect scratches, dents, etc., on various surface textures and can set up the light condition based on the surface so it is ideal for the camera. As shown in image 3.3, the machine consists of a conveyor with a clamping system carrying the inspected part, an inspection unit and industrial robots serving for pick and place tasks. The image 3.4 shows the inside of the inspection unit with a set of cameras. The control loop starts with a robot picking a part from the clamping system, then the images are taken while the part is being moved through a specialised lighting device. After that, the part is placed back in the clamping system and the machine vision software processes the images and decides whether the part is faulty or not. This system utilises deep learning for creating the software logic.

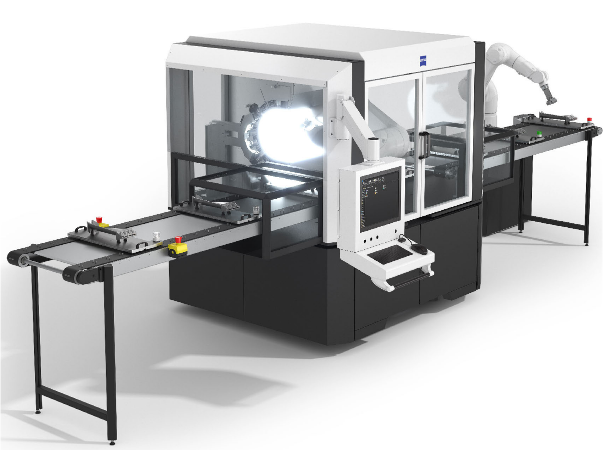


Figure 3.3: Zeiss Surfmax [11]

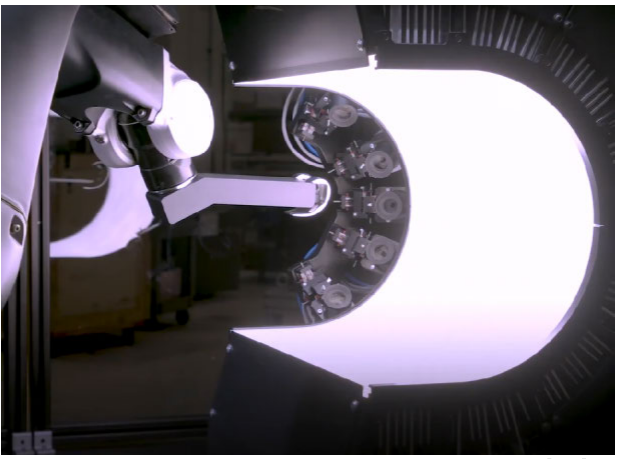


Figure 3.4: Zeiss Surfmax optical system [11]

This Zeiss solution is used for example in the aircraft manufacturing industry. In picture 3.5, you can see the defect detection on the blades in an aircraft engine.

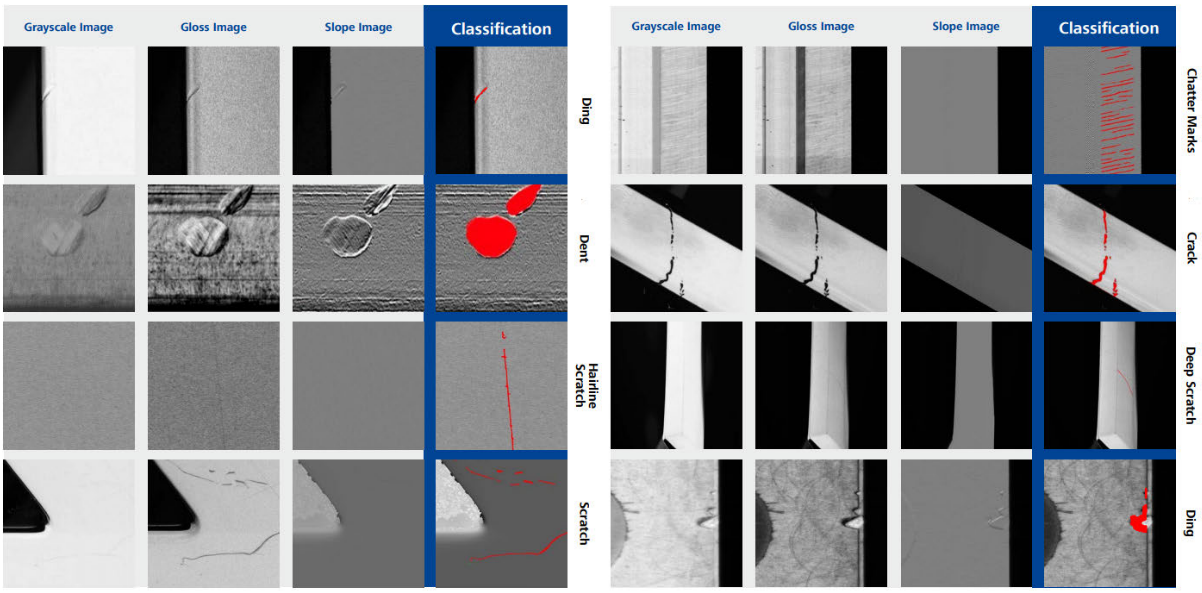


Figure 3.5: Zeiss Surfmax inspection demonstration [11]

Both presented solutions feature concepts with transparent sides of the unit. This design is common in machine vision applications and is advantageous in terms of the required material amount and accessibility. However, it is more difficult to ensure that the light conditions remain constant. This is usually achieved by ring light that creates a light curtain blocking unwanted incoming light from other sources. Zeiss Surfmax is also offered in a non-transparent unit concept. Which concept to choose depends on circumstances.

There are not always just defected and correct parts. There can be an application where sorting into multiple categories is required. The only difference in the software logic is that there is a system that at the end sorts the results into multiple categories instead of just two. During the training of neural network, this makes no real difference. The only requirement for the dataset is that each type of category needs to be labelled.

Another motivation for sorting might be the fact that it generates information about types of defects that can be used for statistical purposes. These statistics might be very beneficial because they create an option to find a systematic error behind these defects and eliminate it. For example, if there is a severe increase of one type of scratch, it can mean a malfunction in the production machine. It can be further distinguished which machine is causing the problem using this information.

A Dutch food processing manufacturer, the company called Marcelissen, developed an optical potato sorting machine [12]. This is a great example of an application where sorting into multiple categories is required. Potatoes of various sizes and defects are fed into the machine and placed in a specialised conveyor. There are four cameras in the closed inspection cell in total, each taking eight photos of a rotating potato. These 32 images are used for classifying the potato. You can see the machine in picture 3.6. The scalable sorting system that comes after the optical unit is an interesting concept. The client can customise the number of categories the potatoes get sorted to and the sorting units together with containers are added accordingly, as shown in picture 3.7.

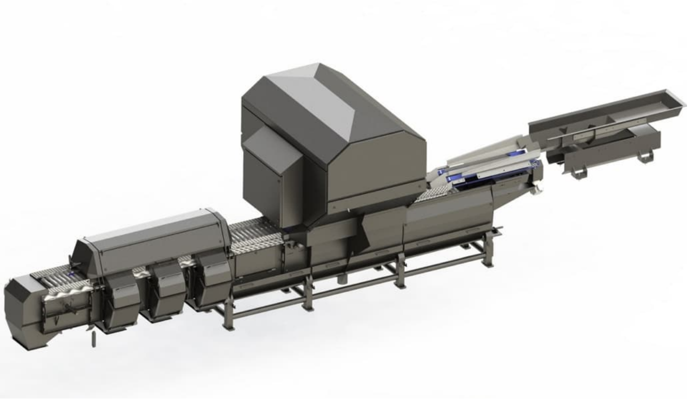


Figure 3.6: Potato sorting machine [12]

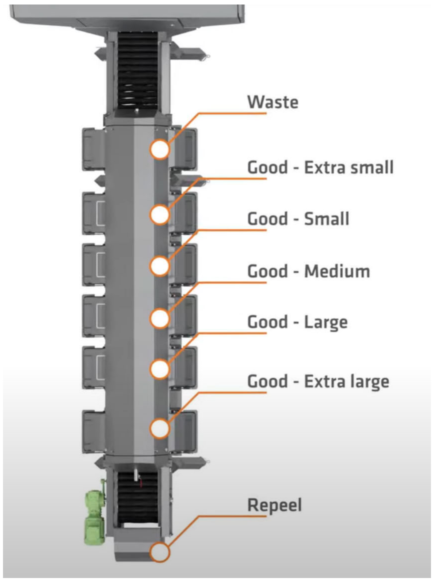


Figure 3.7: Scalable sorting system [12]

3.1.2 Completeness control

Completeness control is becoming a standard part of every assembly machine. There is often a need to check whether all components have been inserted, as it is easy to lose a part [13] during the assembly process, especially in case of assemblies with a high number of parts. There are many reasons why it can happen. For example, there can be a faulty part that does not get attached correctly or the parts magazine gets empty and the sensor breaks down. It can also be caused by a human mistake. Whatever the reason, this problem can lead to some serious damage.

Imagine a situation where there is a small assembly missing a low-cost component such as a spring. This faulty sub-assembly gets installed as a part of a much larger device. As a result, there is a problem during the final testing of the product, which can lead to multiple scenarios. The worst one is marking the whole device faulty, a better one is that it might take a long time to disassemble the device and replace the faulty part, which can in turn lead to some serious financial losses. To minimise the probability of this happening, completeness control using machine vision comes along.

As always, there are various software approaches to this problem. They are generally similar to the ones described in subsection 3.1.1 because completeness control is almost the same task as defects detection from a software perspective. There is simply an unusual element to be detected.

Let us show and analyse some solutions created by machine vision companies. One of the leaders in vision systems, a Japanese company Keyence Corporation, developed a smart camera solution called IV2 [14]. This device uses a neural network to detect irregularities and can be used either for defects detection or completeness control. You can see the application case for steering wheel assembly check in picture 3.8 below. This application checks whether all screws have been inserted. You can see the correct state of assembly in picture 3.9 and a detected missing screw in picture 3.10 This is one of those cases where a lacking screw in the steering wheel might not be discovered even during the final inspection but can have fatal consequences when failing during usage.

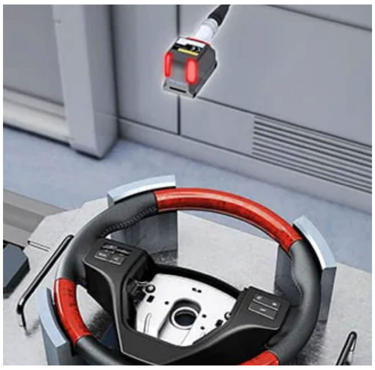


Figure 3.8: Steering wheel inspection [14]

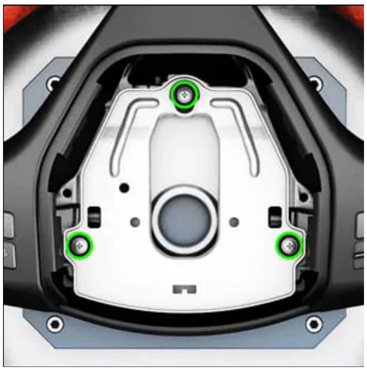


Figure 3.9: All screws placed correctly [14]

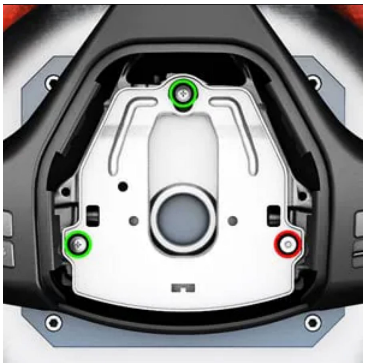


Figure 3.10: Detected a missing screw [14]

The producer of smart cameras, an American company Cognex Corporation, is another leading machine vision company. They offer a wide range of cameras suitable for almost any machine vision application. These cameras use either rule-based machine vision software or artificial intelligence. The pictures 3.11 and 3.12 below show two assembly check applications. The first one is depicting a PCB check. During this operation, the machine vision system checks whether all electrical components are present. The next application shows the detection of wire presence. Both cases use the In-Sight D900 smart camera model, which utilises NN for software recognition [15]. This camera can be also used for defects detection, which confirms the statement that these tasks are similar regarding the software.

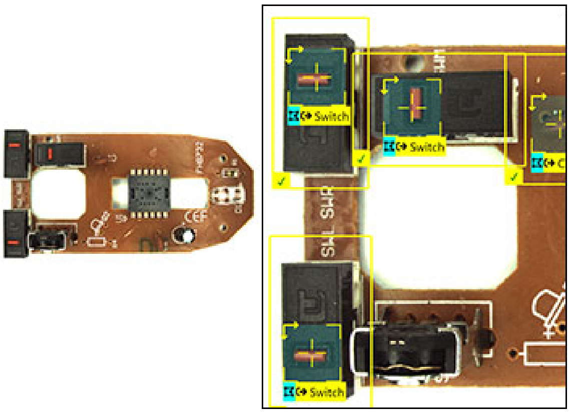


Figure 3.11: PCB assembly check [15]

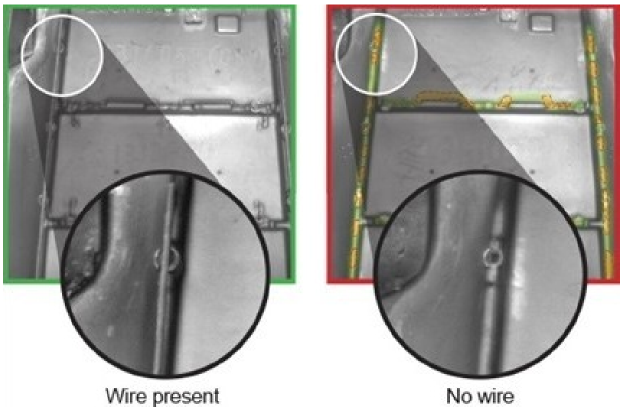


Figure 3.12: Trim assembly check [16]

An interesting application was developed by the company Sensworks GmbH [17] that created a machine vision system for the detection of pralines in a chocolate box. The absence of such control would not have any drastic results. However, its presence raises the chances of delivering a top-quality product. You can see the detection demonstration in picture 3.13.

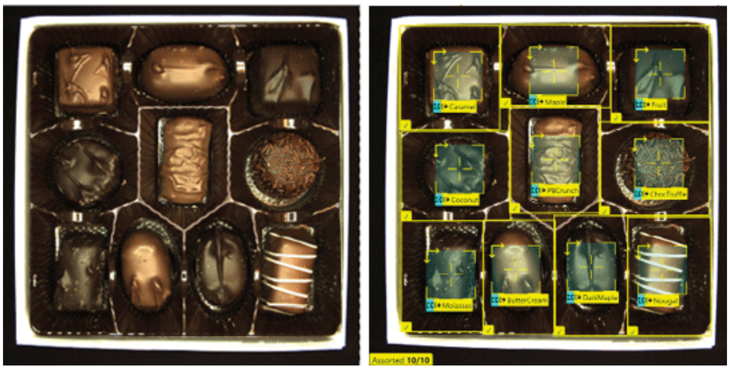


Figure 3.13: Praline box inspection [17]

3.1.3 Contactless measuring

Another type of machine vision application that is different from the ones mentioned in subsections 3.1.1 and 3.1.2 is measuring. Companies are increasingly implementing machine vision systems for measuring as they are much faster and more convenient. One of their advantages is that you can measure many dimensions at once, whereas you generally need to switch between multiple devices when using conventional mechanical methods.

There are various approaches to contactless measuring. You can either use some kind of scanner, cameras with a conventional fixed focal length lens or cameras with a telecentric lens. Which one to choose depends on the precision requirements and other circumstances. Cameras with a conventional lens are usually used for applications not requiring a high precision. The telecentric lens creates a parallel projection of the image onto the camera sensor with only a minimal distortion, which makes it a better choice.

Precision also depends on the camera resolution. Higher resolution generally means higher precision because the size of the pixel dictates the differentiability. Another important factor is how precisely the part is positioned, since when the part is tilted towards the camera, the lengths are distorted during the parallel projection.

In picture 3.14 you can see the Keyence LM series measurement system. It features a double telecentric lens and a 20 MPx high-resolution Complementary Metal–Oxide–Semiconductor (CMOS) sensor [18]. Thanks to the intelligent edge detection it offers measurements to +/- 0.7 μm accuracy and +/- 0.1 μm repeatability.

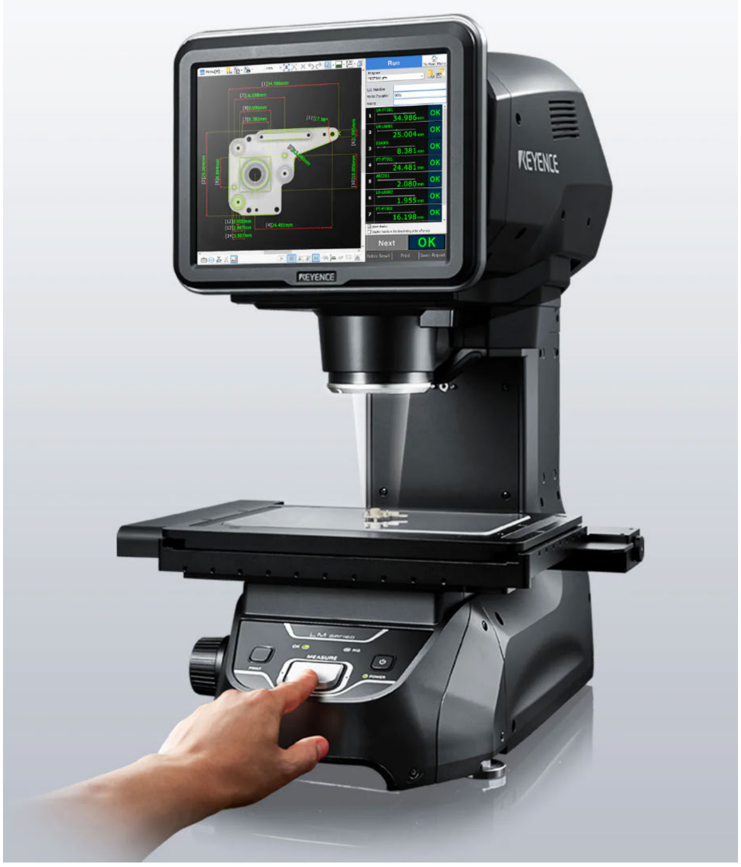


Figure 3.14: Keyence LM series [18]

The picture 3.2 shows the measurement station of AI Inspector One mentioned in subsection 3.1.1. This system utilises telecentric lens together with backlighting.

An interesting solution for 3-Dimensional (3D) measuring is offered by the company ABB. It features a 3D optical scanner that can be attached on a robot, thus creating a system that can scan a part and produce a 3D model with precise dimensions. You can see the device in picture 3.15 and the scanned model with dimensions in picture 3.16.

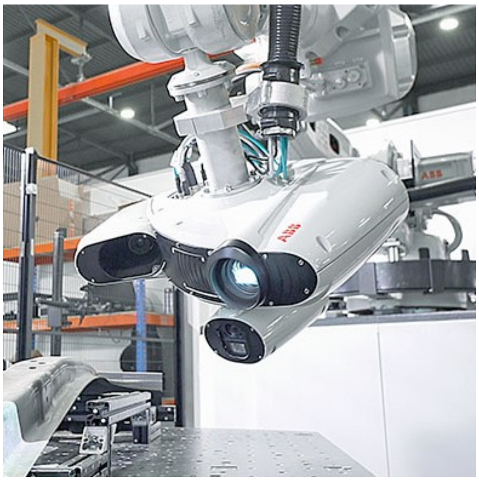
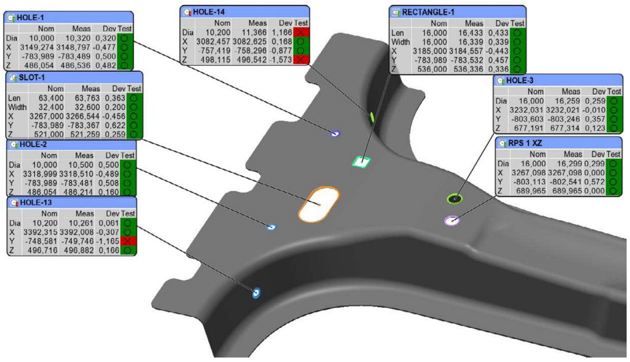


Figure 3.15: ABB 3D Optical Scanner [19]



HOLE-1

	Nom	Meas	Dev	Test
Dia	10,000	10,320	0,320	○
X	3149,274	3148,797	-0,477	○
Y	-783,989	-783,489	0,500	○
Z	486,054	486,536	0,482	○

SLOT-1

	Nom	Meas	Dev	Test
Len	63,400	63,763	0,363	○
Width	32,400	32,600	0,200	○
X	3267,000	3266,544	-0,456	○
Y	-783,989	-783,367	0,622	○
Z	521,000	521,259	0,259	○

HOLE-2

	Nom	Meas	Dev	Test
Dia	10,000	10,500	0,500	○
X	3318,999	3318,510	-0,489	○
Y	-783,989	-783,481	0,508	○
Z	486,054	486,214	0,160	○

HOLE-13

	Nom	Meas	Dev	Test
Dia	10,200	10,261	0,061	○
X	3392,315	3392,008	-0,307	○
Y	-748,581	-749,746	-1,165	✗
Z	496,716	496,882	0,166	○

HOLE-14

	Nom	Meas	Dev	Test
Dia	10,200	11,366	1,166	✗
X	3082,457	3082,625	0,168	○
Y	-757,419	-758,296	-0,877	○
Z	498,115	496,542	-1,573	✗

RECTANGLE-1

	Nom	Meas	Dev	Test
Len	16,000	16,433	0,433	○
Width	16,000	16,339	0,339	○
X	3185,000	3184,557	-0,443	○
Y	-783,989	-783,532	0,457	○
Z	536,000	536,336	0,336	○

HOLE-3

	Nom	Meas	Dev	Test
Dia	16,000	16,259	0,259	○
X	3232,031	3232,021	-0,010	○
Y	-803,603	-803,246	0,357	○
Z	677,191	677,314	0,123	○

RPS 1 XZ

	Nom	Meas	Dev	Test
Dia	16,000	16,299	0,299	○
X	3267,098	3267,098	0,000	○
Y	-803,113	-802,541	0,572	○
Z	689,965	689,965	0,000	○

Figure 3.16: Scanned model with dimensions [19]

3.1.4 Robot guidance

The utilisation of industrial robots significantly improved production efficiency over the past decade [20]. They are used in a wide range of applications across the fields of industry. They are very flexible and compared to a single-purpose machine they are not necessarily fixed to a single application. These robots can thus be used for multiple purposes depending on the current production requirement, which can significantly help smaller businesses.

The first drawback of this seemingly trouble-free product is that it requires an engineer capable of editing the program according to changes. If just a slightly different product arrives, the software needs to be changed. There is also a need to recalibrate the robot each time it is moved.

The second drawback is that in order to precisely manipulate an object, it is necessary to position it, since the robot moves through the points in space and makes some actions. As those points are fixed, when the object is to be manipulated and its position is not identical to the robot's gripping point, it is not grabbed correctly. Furthermore, there is usually no feedback on the correctness of the grip, so a positioning jig or a grid needs to be developed, which seriously impacts flexibility.

A solution to these problems is adding a machine vision system that helps the robot adapt to changes and provides it with information about its environment. This creates room for software engineers to develop a wide range of intelligent machine vision systems processing the image and guiding the robot.

An optical system can be either on-arm, i.e. attached to the arm, or off-arm. Which one to choose depends on the application. The on-arm concept is beneficial in terms of calibration because the coordinate system of the camera is directly fixed to the one of the robot. However, sometimes there is a problem with vibrations and it also makes the end of the robot large. The guidance systems can be divided into 2-Dimensional (2D) and 3D systems. When using 2D systems, all parts are usually placed on the same plane, such as the surface of the conveyor, and do not lie over each other. According to [21], it is cheaper and more common. The 3D system usually employs a 3D camera instead of a conventional camera and in addition it delivers information about depth.

The ABB company offers a 2D vision system which was created in cooperation with the company Cognex and serves for robot guiding [22]. It is a complete solution compatible with other ABB products and is programmable using ABB's programming software RobotStudio. It is very easy to implement as long as used with ABB products. It can be used as an on-arm or off-arm concept. The picture 3.17 below shows an application where the robot is picking variously positioned boxes from a conveyor.

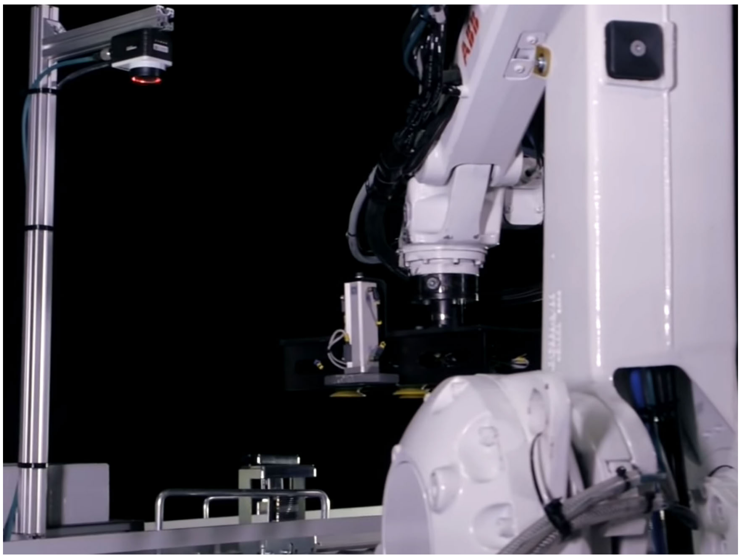


Figure 3.17: ABB vision system for robot guidance [22]

A more demanding operation is 3D bin-picking of parts that are placed over each other in the box. The company Keyence developed a 3D vision system for guiding a robot to pick disorganised items from a container [23]. You can see it in picture 3.18.

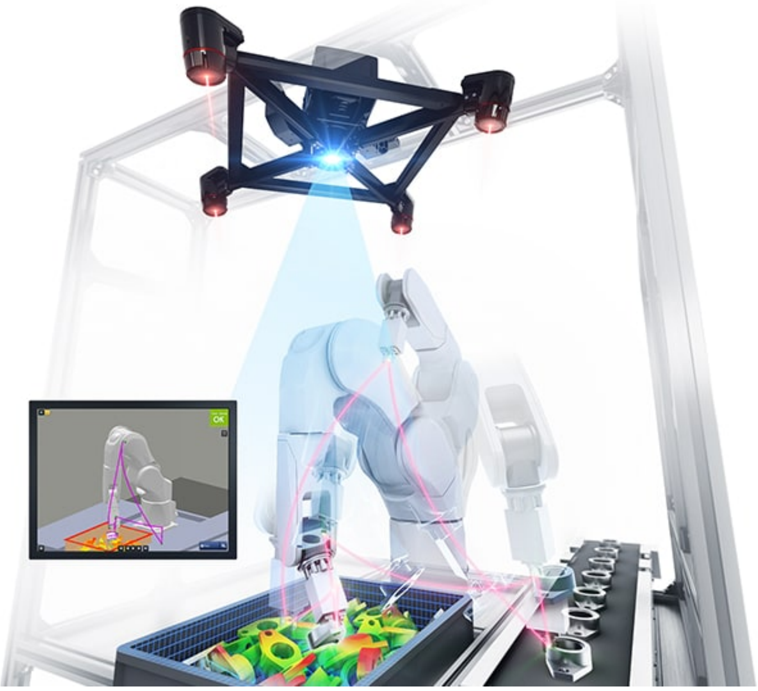


Figure 3.18: Keyence 3D vision system for bin-picking [23]

3.2 Image capturing systems

The aim of this section is to describe optical systems used in machine vision. First, the principles of the cameras are described. After that, the insight into lenses is given and lastly, lighting possibilities are shown.

3.2.1 Cameras

The invention of the camera dates back to the 17th century. Since that time, cameras have been improved significantly. Nowadays, we are able to mass-produce compact high-resolution cameras at a very low price. We can store thousands of images on small devices thanks to digitisation, which also allows image processing. This field of study is concerned with various mathematical methods of transforming and adjusting the image. All of this is possible thanks to the successor of photographic film, which is the digital photographic sensor.

There are two main types of sensors used - the Charge-Coupled Device (CCD) and the more modern CMOS. Both share the same idea of photon trapping which uses light to charge a capacitor. Each pixel has its own capacitor that is charged based on the amount of light during the capture, so we can measure the voltage on it. This voltage is an analogue value that needs to be converted to the digital form using an Analog/Digital (A/D) converter.

This is where the main difference between CCD and CMOS technology lies. As the CCD has usually only one A/D converter, the voltage from only one pixel is read at a time. The CCD thus only has one system of additional electronics for the whole pixel array. On the other hand, CMOS has additional electronics in each pixel, as shown in pictures 3.19 and 3.20. This means that the pixel array output is already a digital matrix. This is possible thanks to the advancement in photolithography, which allows manufacturing extremely small electronic parts.

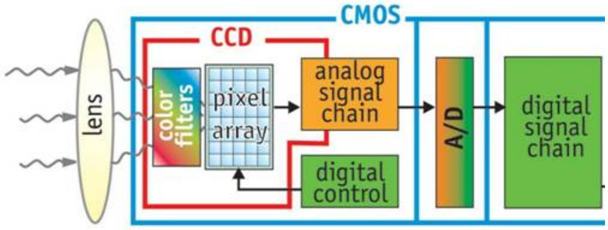


Figure 3.19: Electronics of CCD and CMOS [24]

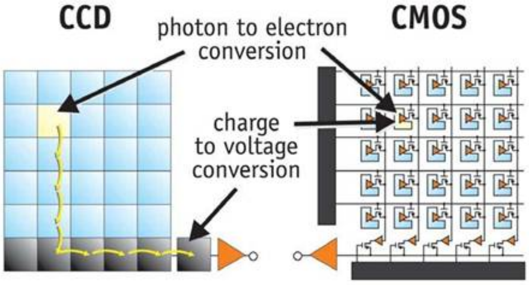


Figure 3.20: CCD vs. CMOS sensor [24]

Pixels need to be illuminated for a defined time, which is called exposure or shutter time, and then during the A/D conversion we have to make sure they are no longer exposed. The device responsible for it is called a shutter. It can be either a mechanical or electronic device that creates a certain light curtain.

The mechanical variant is better from the image quality perspective but it has some downsides. Firstly, it is quite a complex device that has to endure extreme acceleration and might get broken. The fastest mechanical shutters are able to open or close in 1/8000 of a second. Secondly, the rapid acceleration and deceleration create vibration, and lastly, they require a lot of space. There are two main designs of mechanical shutters that you can see in pictures 3.21 and 3.22. The left one is a leaf shutter design and the right one is a focal plane shutter.

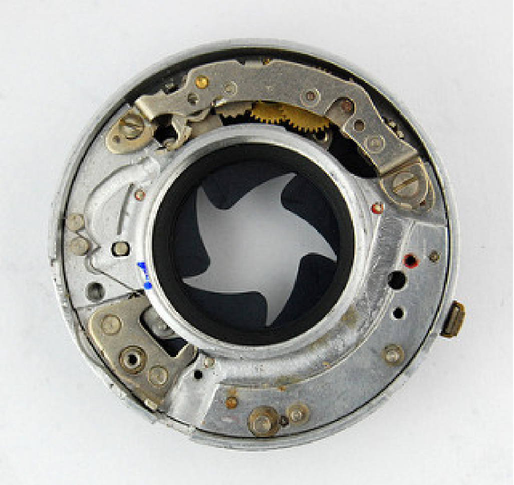


Figure 3.21: Leaf shutter [25]

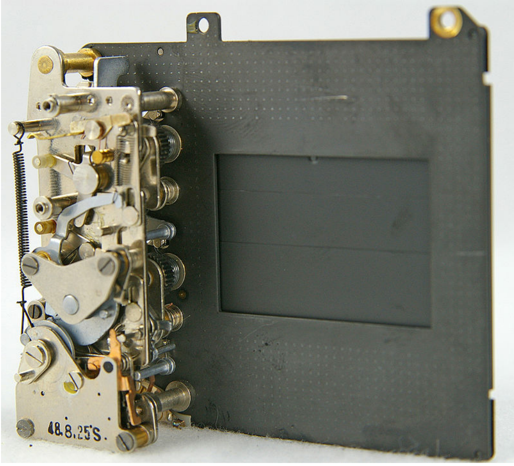


Figure 3.22: Focal plane shutter [26]

The electronic shutter has no moving parts at all. Instead of covering the image sensor it simply resets each pixel and then reads it after a defined time. This technology is already a part of the sensor. There are however a few problems. One of them is that these sensors generally have a higher image noise. Another drawback is that the data from all the pixels cannot be read at once, but line after line instead, which creates a distortion. Let us explain.

Shutter technology falls into two categories, global shutters and rolling shutters. We refer to the global shutter when all the pixels are exposed at the same time, whereas we talk about the rolling shutter when all the pixels are not exposed simultaneously and the exposure of each line of the pixels starts at a different time. You can see the difference demonstrated in pictures 3.23 and 3.24.

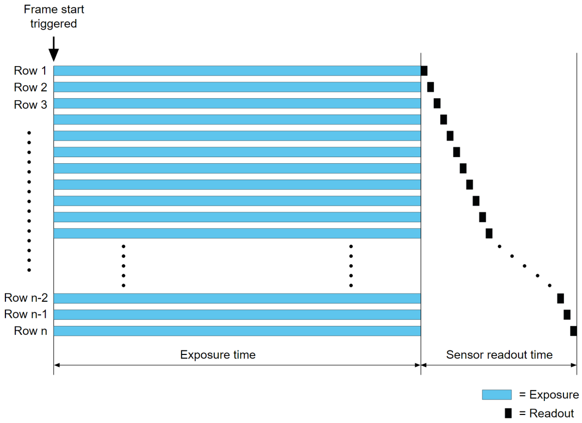


Figure 3.23: Global shutter [27]

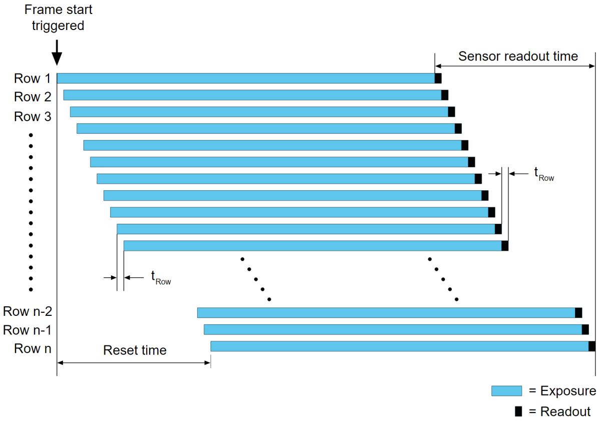


Figure 3.24: Rolling shutter [27]

One can say that the global shutter is typical for mechanical variants and the rolling shutter is typical for electronic ones, but it is not a rule. In general, it is always better to have a global shutter. Rolling shutter is a technological compromise. The reason why it is not as good is that the final picture is composed of lines of pixels that were taken at different times. Even though the time difference between the first and the last line is extremely small, there is a possible distortion when capturing a rapidly changing scene, such as a fast-moving object. The difference between the same image taken by a camera using a global shutter and by a camera with a rolling shutter is illustrated in picture 3.25.

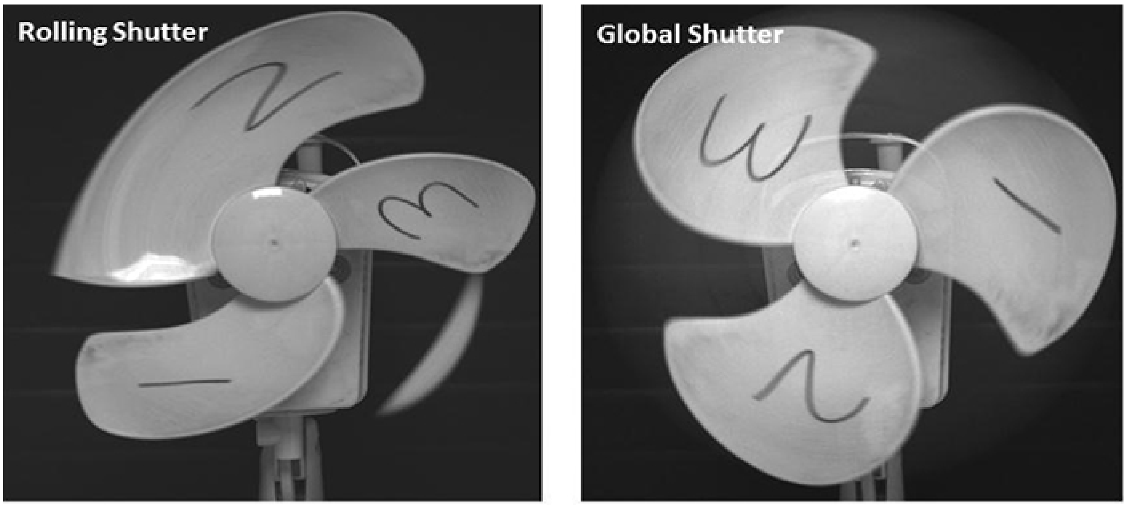


Figure 3.25: Rolling shutter image distortion [28]

Sensors using electronic rolling shutters are widely used for example in smartphones because they are cheap and the possible distortion is usually not an issue. However, they are quite problematic when used in industrial applications where a precise image is required. Using a mechanical shutter would be also very problematic, as these cameras are often used in a non-stop mode and the shutter would wear out quickly. Fortunately, companies have come up with modern CMOS sensor solutions with electronic global shutters. The leading sensor manufacturer, the company Sony Corporation, developed a new sensor line using the global shutter technology called Pregius. Details about this technology can be found here [29]. It basically resets the whole sensor and then saves the pixel data in memory all at once before it is processed. A similar solution was created by the company ON Semiconductor and is called Python series sensors [30].

Let us say we have our pixel array charged and we convert it to a digital matrix where each element represents the brightness of light that was coming to the pixel. These elements may range depending on the sampling. We usually sample the spectrum to a 8-bit value, meaning the lowest number is 0 and the highest one is 255. These numbers can represent the shades of grey, where 0 is black and 255 is white. We can now display the image. However, it would only be in greyscale as we get absolutely no information about the colours.

To get a colour image, we need to separate the light into wavelengths representing certain colours, illuminate pixel arrays with a separated light of a certain wavelength and then combine them together. This way we get multiple matrices, each representing brightness of a certain colour. The most common is a combination of three matrices, standing for the red, green and blue colour, known as RGB.

To achieve this, multiple technologies have been developed. The most straightforward one is called 3CCD [31]. It uses a beam-splitter prism that splits the incoming light into three beams, each representing a different colour. This technology is nevertheless large and expensive because it requires three sensors. It is thus not commonly used. You can see it in picture 3.26.

Another interesting technology was developed by the company Foveon, Inc [32]. It uses three layers, where each is able to absorb a certain wavelength representing colour. The technology is shown in picture 3.27. The main reason why this technology is not widely spread is that the company is not willing to allow other companies to use the patent.

The most common technique is a Bayer filter mosaic [33]. It utilises a grid of the same size as the sensor that has segments that only let pass the light of a certain wavelength or colour. As a result, a pixel receives only red, green or blue light. The usual distribution is 50% of pixels receiving the green light, 25% red light and 25% blue light. This is because the green colour is the most common one. The downside of this technology is that it reduces the resolution of the final image as it uses only a part of the sensor pixels for each colour. You can see the picture 3.28 showing the Bayer filter below.

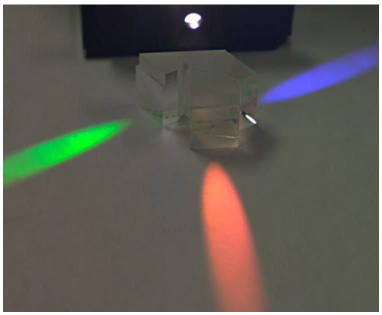


Figure 3.26: 3CCD [31]

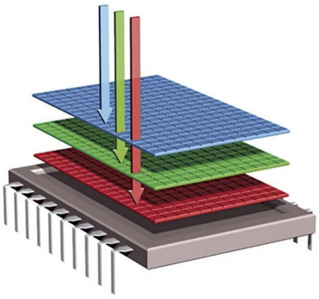


Figure 3.27: Foveon X3 [32]

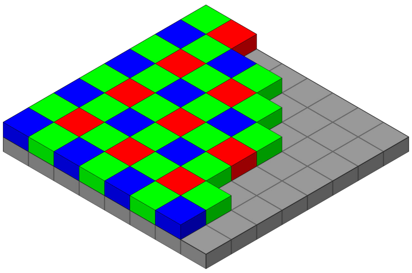


Figure 3.28: Bayer filter [33]

Industrial cameras differ from the common ones in smaller size, they usually do not have their own power source and data storage, they are designed to work non-stop, do not have any optics or display, and are usually controlled remotely. There are a lot of manufacturers who offer different variations designed for a specific task. The basic parameters are the resolution as well as the size of the sensor, whether it is a colour or greyscale camera, the maximum Frames Per Second (FPS) and what port it uses. In essence, the higher the resolution and the size of the sensor the better. However, a higher resolution also means more data and more demanding post-processing. It needs to be considered whether a high resolution is necessary for a particular application. High FPS cameras are required in applications where capturing fast-moving objects is needed. The camera is usually controlled and powered either by Universal Serial Bus (USB) or by Ethernet, in which case Power Over Ethernet (POE) is used.

Let us have a look at some industrial cameras from leading manufacturers. Each of them offers a wide range of products and they all use sensors from manufacturers like the Sony Corporation. Therefore, the differences are not that great. The main criteria when choosing the right one are the price, quality of build and probably the most important one, the availability. In times of writing this thesis, the availability of cameras is greatly reduced due to the global shortage of semiconductors. The picture 3.29 shows a basic model by the manufacturer Basler, the picture 3.30 shows the Jai camera and picture 3.31 a camera by the Flir company. All these models use the same sensor and the design differences are noticeably rather insignificant.

Figure 3.29: Basler camera [34]

Figure 3.30: Jai camera [35]

Figure 3.31: Flir camera [36]

3.2.2 Lenses

In order to concentrate the light and project the image on the sensor, an additional device needs to be attached. It can either be a mirror system or a lens system. The mirror system is used extremely rarely in industrial applications.

There are multiple types of lenses and each of them suits different applications. The most basic division is between the ones with a parallel projection and ones with a perspective projection. A lens with parallel projection has an infinite focal length which means the light rays entering and exiting the lens are parallel. Thanks to this, the size of an object in the photo does not depend on the actual distance of the object from the camera. There is no fish-eye distortion either. The lenses with perspective projection are more common and they usually have a fixed focal length. When the actual distance is increased, the object in the image appears smaller.

The main type of parallel projection lens used in the industry is the telecentric one. These lenses are used in applications where high precision is required or where the object is not precisely positioned. The main reason is that the fish-eye phenomenon, which is present when using perspective projection lenses, increasingly distorts the lengths with the object's distance from the image centre. This can bring imprecision to the measurement. However, it does not occur with a telecentric lens. Furthermore, the pixel to real length ratio does not change as it does with the perspective projection.

Why would then anybody use a lens other than telecentric? The main reason is that the telecentric lens has a fixed Field of View (FOV), so you can only see the same size area as the lens entrance area. Therefore, if it is required to see a large area, the lens has to be large as well. It is typically not only wide but also long. You can see a comparison of the conventional and telecentric lenses in picture 3.32 below. There are some special types of the telecentric lenses shown, but the both-sides telecentric lens is the most common one.

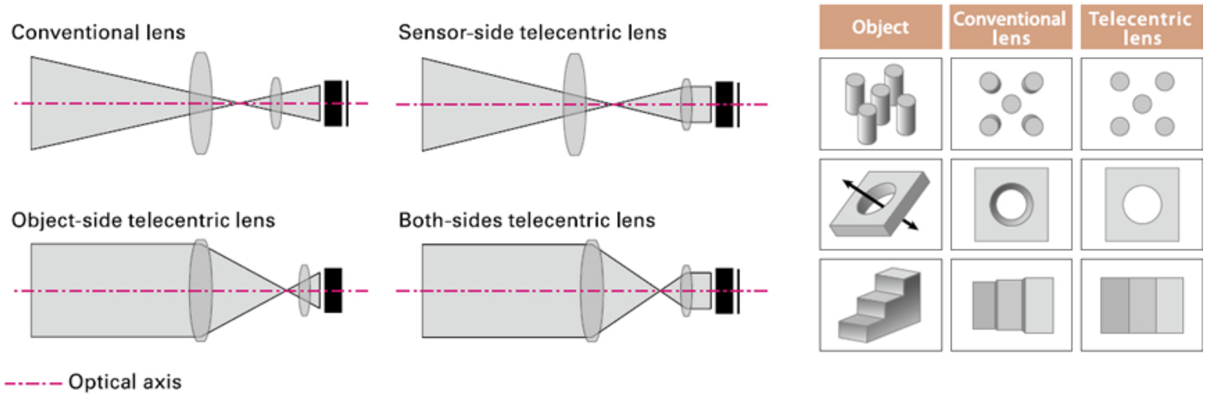

Conventional lens
Sensor-side telecentric lens
Object-side telecentric lens
Both-sides telecentric lens
Optical axis
Object
Conventional lens
Telecentric lens

Figure 3.32: Conventional vs telecentric lens [37]

There are more categories of lenses with the perspective projection. The ones with a fixed focal length (f) are the most commonly used in industrial applications. Their main characteristic is that they have a fixed Angular Field of View (AFOV), which defines the cone of projection coming into the lens. The smaller the focal length, the bigger the AFOV, which leads to a bigger FOV at the same distance.

Furthermore, they have a variable focus, meaning that there is a possibility to adjust the focus depending on the Working Distance (WD), which is the distance between the object and the lens. However, to achieve a sharp image there is an optimal range of WD. It is for instance not possible to focus on the object that is in very close proximity to the lens. The manufacturer usually offers an online lens calculator, where you compute the parameters, such as here [38]. You can see the difference between a longer and shorter focal length in picture 3.33, where H is the horizontal size of the camera sensor.

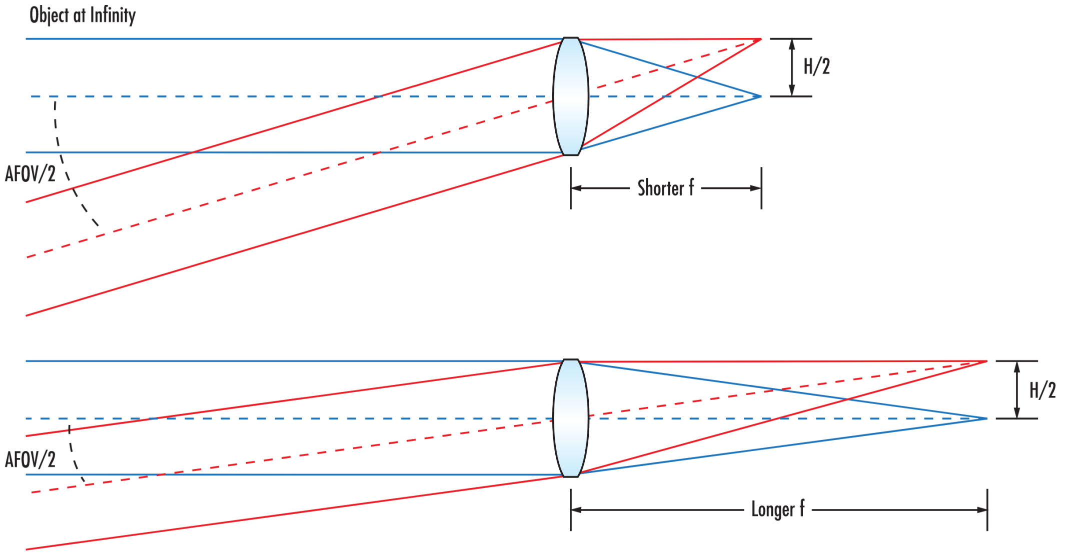


Figure 3.33: AFOV dependency on focal length (f) [39]

It is clear from picture 3.33, the AFOV can be computed using this equation 3.1:

$$AFOV = 2.tan^{-1}(\frac{H}{2f}) \tag{3.1}$$

However, there is usually a need to compute FOV based on the WD and choose the lens with the correct f. Therefore, the equation 3.2 is handy.

$$FOV = \frac{H.WD}{f} \tag{3.2}$$

The situation is demonstrated in picture 3.34 below.

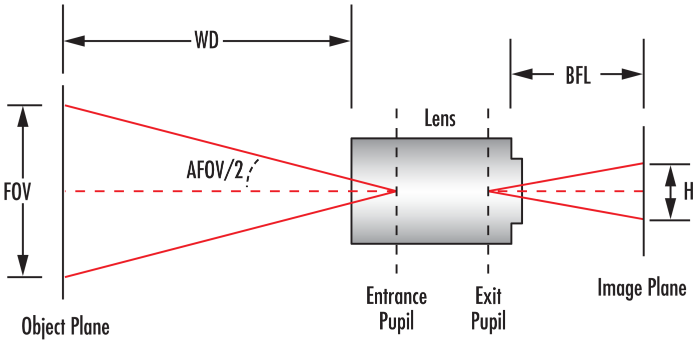


Figure 3.34: Fixed focal length lens [39]

The focus is usually adjusted manually and then fixed for the operation. There are also remotely controlled lenses that use a liquid to focus [40].

Sometimes there is an application where imaging of objects of different sizes or distances is required. Zoom lenses are ideal for such applications because they have a variable focal length and therefore have adjustable FOV without the necessity to change WD. They are not used in industrial applications very often as they are usually large and heavy. They are nevertheless often used by photographers. The difference between the lens with a fixed focal length or prime lens and the one with zoom can be seen in picture 3.38.

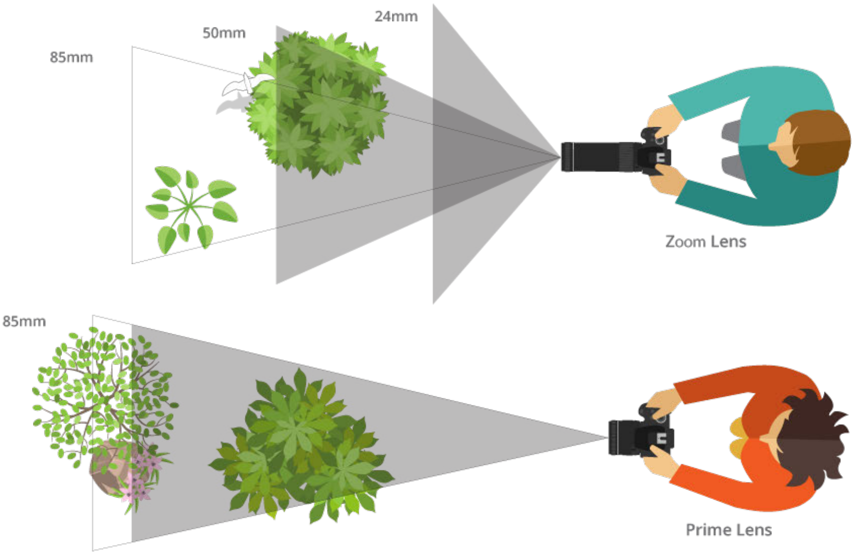


Figure 3.35: Fixed focal length lens [41]

Let us have a look at some products from the manufacturers. One of the leaders is the American company Edmund Scientific Corporation. They offer a variety of models from each category mentioned. You can see typical models in pictures 3.36, 3.37 and 3.38.

Figure 3.36: Telecentric lens [40]

Figure 3.37: Fixed focal length lens [40]

Figure 3.38: Zoom lens [40]

3.2.3 Lighting

In order to achieve a high-quality picture with highlighted elements of interest, proper lighting is essential. Lighting can be divided based on placement or design. Each category is useful for a different type of application.

When measuring or inspecting the outline of the part, the most used method is backlighting. It can be achieved by placing the object between the light source and the camera. Sometimes the object can lie directly on the light source as you can see in picture 3.39. The main advantage of this method is that it achieves a high contrast between the object and the background and makes segmentation and edge detection easier.

Other commonly used types of lighting are the dark-field and bright-field illumination. These two methods both use a light source that shines directly at the object and the camera absorbs the reflected light. The difference between each method is that the bright-field illumination shines at the object from the direction close to the camera, as shown in picture 3.40. This way is better for high-contrast objects, such as PCB. On the other hand, the dark-field illumination uses lights that illuminate the part more from the sides. This way the defects, such as scratches, come up. There can nonetheless be a problem with shadows. For this reason, there are usually two light sources from each side, as shown in picture 3.41.

Sometimes there is a need to create a diffused uniform light that does not create reflections or shadows. This type is useful for example for colour surface control. There are multiple possibilities to achieve this but indirect lighting is typically used. This can be done by reflecting the light from a matte uniform surface. You can see the concept using a reflective dome in picture 3.42.

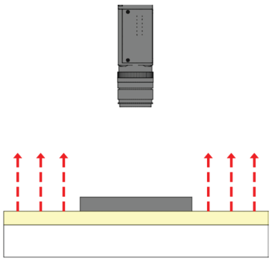


Figure 3.39: Back light [42]

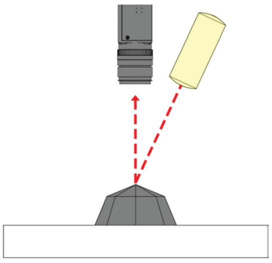


Figure 3.40: Bright-field light [42]

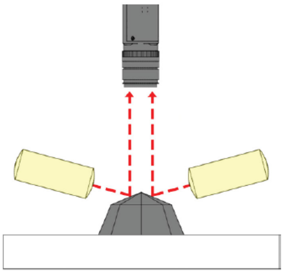


Figure 3.41: Dark-field light [42]

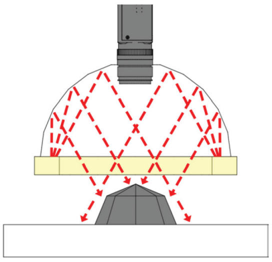


Figure 3.42: Dim light [42]

There are also additional special methods of lighting, such as using a semi-transparent mirror or creating a structured light. But these are used only for special purposes.

Let us look at some designs of industrial lights. The vast majority of commonly used lights have Light-Emitting Diode (LED) as a light source. They are variously distributed and covered. There is usually a planar concept with the surface made of diffusive plexiglass for backlighting. It is designed so that the object can lie on the top of it. An example of this product is shown in picture 3.43.

A design that is very frequent is called a ring light. It has a ring shape. LED lights are distributed in a circle and there is a hole in the middle for a camera. This design is advantageous because the object is evenly illuminated from all sides and there is room for a camera. Manufacturers offer variants with different angles of shine. Therefore, even though it is mostly used for the bright-field illumination, there are models that shine more from the sides as well. The ring light by the company Wordop is shown in picture 3.44.

Another very common concept is a bar light. It can be fitted in various positions in order to produce a certain type of light. It is nevertheless usually used either for the dark-bright illumination, when it is attached so that it shines at the object from the sides, or it can be pointed at the reflective surface to produce a dim light. You can see an example in picture 3.45.

Figure 3.43: Backlight [43]

Figure 3.44: Ring light [43]

Figure 3.45: Bar light [43]

To create a dim light, a dome design is offered. It creates a uniform light. You can see one in picture 3.46. There may sometimes be a requirement to create a concentrated narrow beam light to illuminate only a certain area of an object. Spot lights were developed for this purpose. Such a light is shown in picture 3.47.

Figure 3.46: Dome light [43]

Figure 3.47: Spot light [43]

The type of power source for the lights is also very important. Generally, there are problems with low-frequency unstabilised PWM voltage sources, as they can interfere with the camera and create flickering or banding. To eliminate that, manufacturers offer either stabilised power sources where the voltage shape is not wavy or those with changeable PWM frequency, so they can be set up not to interfere.

3.3 Convolutional neural networks

This section describes Convolutional neural networks. It starts with basic principles, following with an insight into the structure. The function of the layers design is described here. After that, open-source CNN architectures along with pre-trained weights are analysed and compared. Frameworks used for working with NNs are shown and described at the end of this section.

3.3.1 Basic principles of NN

A neural network is a mathematical apparatus that is categorised as an artificial intelligence method. The reason why it is called a neural network is that the idea behind it is to simulate the work of neurons in the human brain. However, to state that it works like a brain would be exaggerating. NN is composed of neurons. The idea is that these neurons are set into layers and that the ones from each layer are connected with the ones in the neighbouring layer. This way a network is made. You can see the basic design in picture 3.48. The first layer is known as the input layer and has a defined size corresponding to the size of the input. Others are called hidden layers and can have various sizes. The L_2 and L_3 that you can see in picture 3.48 are called fully connected layers. The last is called the output layer.

Each neuron has a certain design. The most usual one can be seen in picture 3.49 and consists of inputs that are usually numerical values and weights that are coefficients changing during the training process. After the multiplication of inputs and weights, these values are summed up and the outcome serves as an input to activation functions. Different functions can be used. They need to be nonlinear, monotonous and derivable. Probably the most common one is the sigmoid function, which you can see in picture 3.50. The output is therefore a single value ranging from 0 to 1 that serves as an input to neurons in another layer. Other commonly used activation functions are hyperbolic tangent or ReLU function. By decreasing the number of neurons in each layer, the size of the input vector to upcoming layers decrease as well.

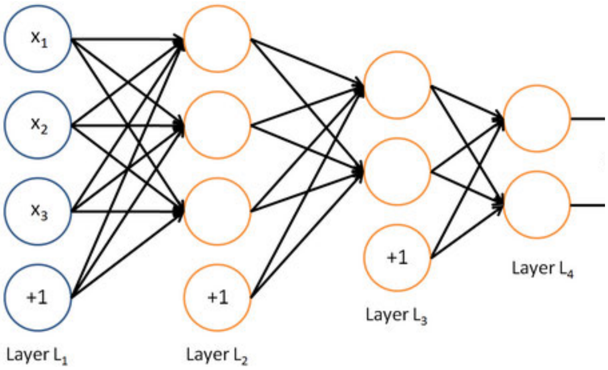


Figure 3.48: Basic NN design [44]

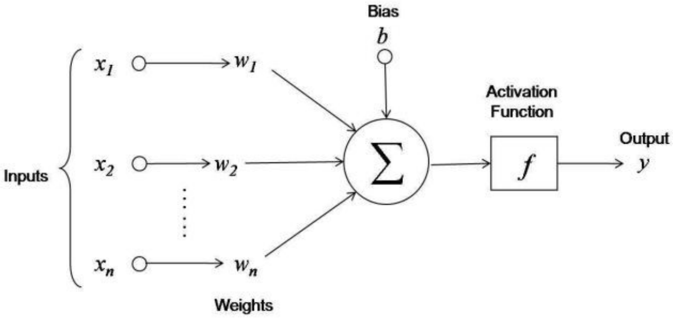


Figure 3.49: Neuron build [44]

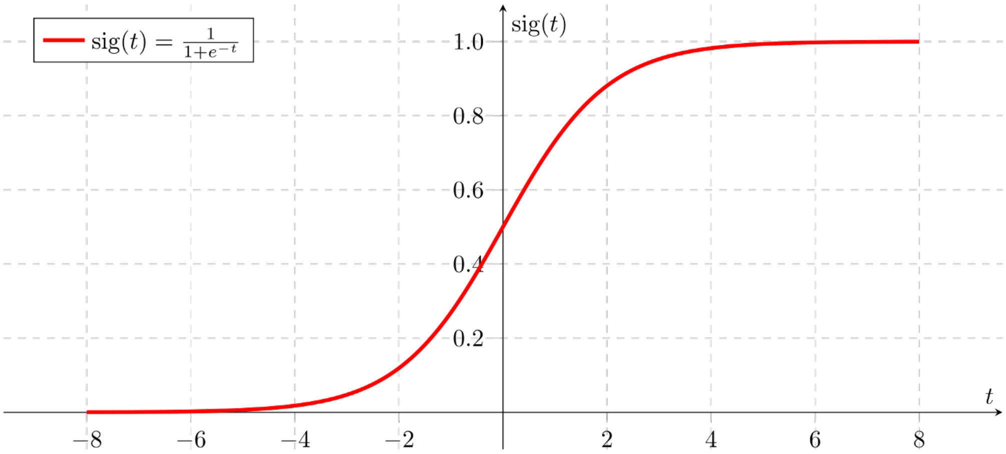


Figure 3.50: Sigmoid activation function [44]

The main idea behind this whole mechanism is that the weights in each neuron are adjusted so that the desired output is generated based on input. The type of output can vary. It might be for example only a true/false value, it can give a score, classify or predict.

The process of adjusting the weights, so that the NN works and does not generate a random output, is called training. During this process, data with the desired output are fed into the NN. After that, the weights are stepwise adjusted based on the difference between the desired and real output. The mathematical process hidden behind it is rather complex.

To put it simply, after the result comes out, it is compared using the loss function, which represents the dependency of weights on the difference between the actual and desired value. Then the direction to change each weight is found using a gradient of the loss function (3.6). This process is called backpropagation and is based on the gradient descent method, which can be improved by adding a momentum or using the Levenberg-Marquardt method.

After that, the weights are adjusted according to the direction of change and size of the difference. This procedure is repeated several times until the NN gives satisfactory results. There are different loss functions that can be used, such as the mean square error (3.3), negative log likelihood (3.4) or cross-entropy loss function (3.5). You can see the equations for these functions and gradient below, where $\mathrm{E_k}$ is a loss function, $\mathrm{y_{kj}}$ is a certain desired value, $\mathrm{o_j}$ is the output value of a certain neuron, ∇ is the gradient and $\mathrm{w_{ij}}$ is a certain weight.

$$E_{\mathrm{k}} = \frac{1}{2}\sum_{j=1}^{M}(o_{\mathrm{j}} - y_{\mathrm{kj}})^2 \tag{3.3}$$

$$E_{\mathrm{k}} = -\sum_{j=1}^{M} y_{\mathrm{kj}} ln(o_{\mathrm{j}}) \tag{3.4}$$

$$E_{\mathrm{k}} = -\sum_{j=1}^{M} y_{\mathrm{kj}} ln(o_{\mathrm{j}}) + (1 - y_{\mathrm{kj}}) ln(1 - o_{\mathrm{j}}) \tag{3.5}$$

$$\nabla = \frac{\partial E_{\mathrm{k}}}{\partial w_{\mathrm{ij}}} \tag{3.6}$$

Let us dive more into the possible types of output. There are two main categories of NNs based on the task type. The first one is classification, which at the output gives the probability of the input falling into each category. There can be a theoretically unlimited number of categories. An example of this type of task is that the NN predicts the probability of a person being a man or woman based on anthropometric measurements such as height, weight, BMI, etc. If we had a dataset with these values with a corresponding label, we would be able to train the network so that it could predict a person's sex based on the input.

The second category is regression, which is not giving a probability of the input belonging to each class but predicts a certain value typical for the input data. An example might be that we give the NN anthropometric measurements and want to predict another one, such as the size of foot. Again, there must be a dataset with the corresponding size of foot used for training.

It is clear that the data used for the training process are essential. In order to create a well-functioning NN, the dataset must be diverse and uniformly distributed. Otherwise, some unwanted effects might occur. The most common problem is called overfitting. This problem is caused by the fact that weights are adjusted so that the NN perfectly predicts data used for training. However, this often leads to bad predictions on data that are just slightly different. As you can see in picture 3.51, the graph on the left shows the curve that goes perfectly through the points. Yet if we wanted to guess the value between the marked points, we would get an insensible output, because the curve does not represent the trend of data. The correct fitting is represented by the middle graph. Another extreme is underfitting, which is illustrated in the graph on the right. This happens when training is insufficient and weights are not yet adjusted to predict the desired output.

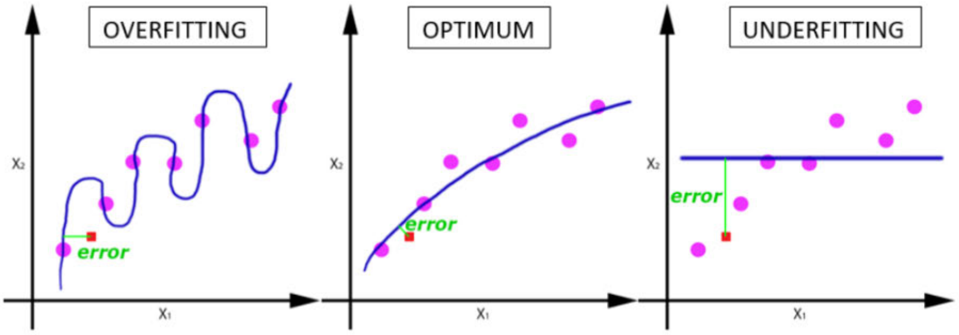


Figure 3.51: Illustration of potential training problems [45]

There are several tools to avoid this from happening. Some of them are used to pre-process the dataset, others are part of NNs architecture. First, let us talk about the pre-processing ones. The first important thing to do to your dataset before it is used for training is normalization. This means that the data values are scaled down to a range between 0 and 1, sometimes -1 and 1, depending on the activation function. The reason for this is obvious when you look at the sigmoid curve in picture 3.50. It is clear that this function is sensitive to the input in a range between -4 and 4. On the other hand, even though there is a huge difference between two high-value numbers, it makes no big difference in the output of the function. The data are therefore scaled-down, so that the difference between each input in the training dataset creates a proper reaction.

Another important thing is to shuffle the data. The reason for this is that the training data are usually sorted into categories. If we used them directly, data for each category would be used sequentially. This way the first category would be underfitted and the last category overfitted. The idea is to shuffle the data so that during training there are diverse data used consequently.

Sometimes, there might be a problem that we are unable to obtain a sufficiently diverse dataset. For this situation, there is a method that can be used to create new data based on the old ones. This is called data augmentation and it uses certain mathematical operations to slightly change existing data but to preserve the same character. This can be used only in certain cases and the way to do it depends on the type of dataset.

Architecture can be also adapted to prevent these problems. The main way to do this is to use dropout. This technique sometimes skips randomly chosen neurons during training. This changes the NN during training and prevents it from overfitting.

The precision of prediction capability of the trained neural network must be measured using the data that were not used directly for training. Therefore, the common practice is to separate part of the dataset and use it for validation. Sometimes when referring to training, the terms machine learning or deep learning might be used. The terms deep neural networks and deep learning are more of a marketing expression because they only state that there is a large number of hidden layers. Furthermore, there is no clear border between an average NN and a deep NN.

3.3.2 CNN and its structure

A convolutional neural network is a variant of a neural network that has images as an input. It uses the mathematical operator convolution to process the images, therefore the name convolutional neural network. You can see the illustration of CNN's architecture used for classification in picture 3.52. The main difference is that there are multiple convolutional and pooling layers preceding a classic neural network, which is represented under the classification label in picture 3.52. Everything that was described in subsection 3.3.1 applies to CNN all the same.

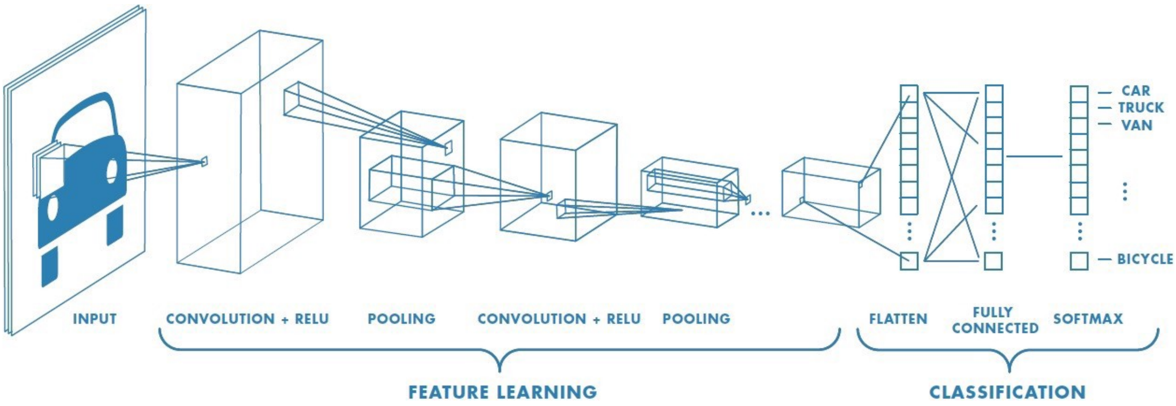


Figure 3.52: Illustration of CNN architecture [46]

Let us look at the build of these layers in detail. Convolution is a mathematical operation. It can be applied to a 1-Dimensional (1D) vector of data, which is used in signal processing. It can be applied to 2D objects as well, where we use the kernel, which is a 2D matrix that slides over the image matrix and performs convolution. This way a new matrix is generated, which contains values that were created based on the interaction between the kernel and the elements in the matrix, which represent pixels. Thanks to that we are able to distinguish certain aspects of the image based on these values. The kernel used for this operation acts the same way the weights did. This means that we are able to set the kernel values so that when it interacts with a typical aspect of the image, a certain value is created. After that, each element serves as an input to the activation function, usually Rectified Linear Unit (ReLU). This way the new matrix of the same size is made with each element being the output from the activation function.

To further extract the main aspects and scale down the matrix, a pooling layer is used, which creates a smaller matrix by extracting the highest value from a region of a matrix coming from the convolution layer. You can see the illustration of this process in picture 3.53 below. This layer is not changed during training. It is not a fully-fledged layer, but more of a tool to scale down the matrix. Usually, there are multiple convolutional layers preceding a single pooling layer.

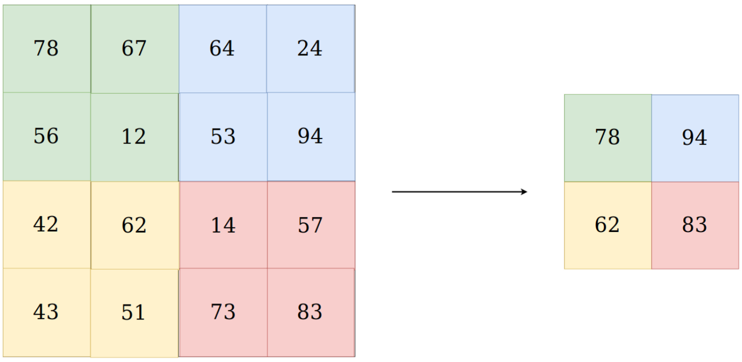


Figure 3.53: Function of pooling layer [47]

There are usually numerous convolution and pooling layers creating a relatively small size matrix at the end. Elements of this matrix represent the characteristic aspects of the image. These values serve as an input to the classic neural network for classification or regression tasks, which was described in subsection 3.1.1. But before this happens it is required to change the 2D matrix into a 1D vector. This operation is called flattening and is very straightforward. Elements are simply put into the vector one by one starting from the upper left element and going row by row. An illustration of this process can be seen in picture 3.54.

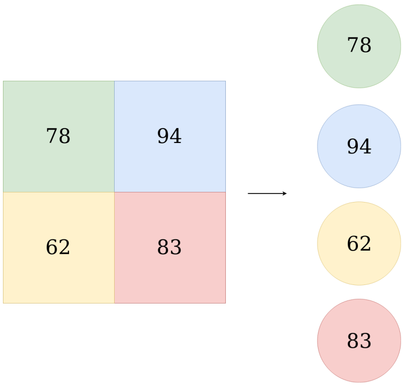


Figure 3.54: Flattening operation [47]

At the end of the NN is a layer with an activation function depending on the type of the task. There usually is a softmax activation function used for classification and a linear function used for regression. A typical regression task is predicting the position of an object in the picture. The output of such CNN might be for example the position of points defining the frame surrounding the object.

Training of CNN is from the mathematical point of view similar to the training of classic NN and all the rules mentioned in subsection 3.1.1 apply all the same. The difference is that it is generally much more demanding on computing power. Graphic cards are designed to process images and execute operations like convolution much faster than the processor. Because of that, graphic cards are often used for training as well as final predicting. Images must preserve the same size as it was with input data to classic NN and are usually divided into batches, which is an object containing multiple images. This mechanism divides the dataset into smaller groups to generate a less memory demanding situation.

3.3.3 Practical use of CNNs

When using CNNs in practice, it is very rare to design the whole architecture from the scratch. Major tech companies like Alphabet Inc. or Meta Platforms were, thanks to their access to the unparalleled amount of data, able to develop open-source architectures that can be used. These CNNs give very good results and it would be almost impossible to create a network that would be as good. Furthermore, these networks can be obtained with pre-trained weights, which were trained using a diverse dataset, such as ImageNet [49], featuring over 14 million images sorted into more than 20 thousand categories. Another example of such a dataset might be Places365 [50], which is a dataset specialised to scene classifying, containing over 10 million images divided into 365 categories.

The convolutional layers are pre-trained to detect aspects of interest in images. A user usually edits the last layers of pre-trained CNN to serve his application. The idea of using these networks is depicted in picture 3.55

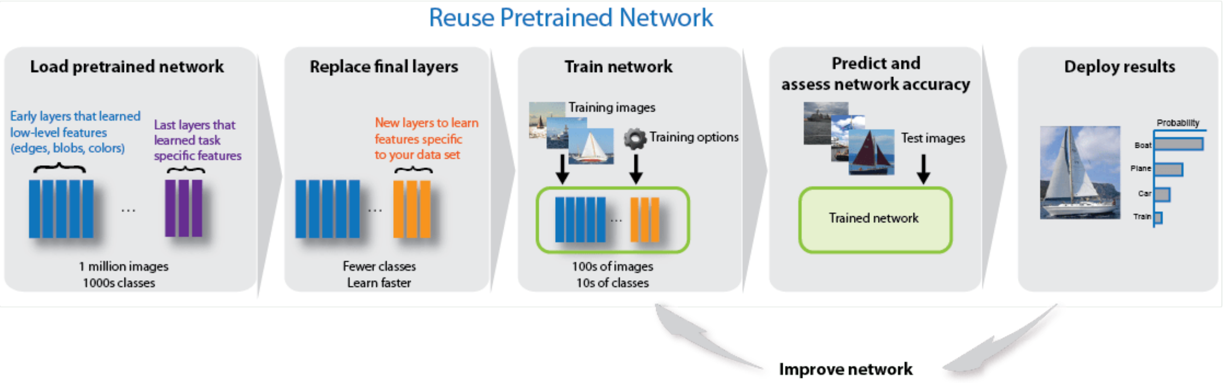


Figure 3.55: Idea of using pre-trained networks [48]

As was already said, there are multiple open-source CNNs available. Their number keeps rising as there are deeper networks with higher accuracy created, thanks to the advancement in electronics components enabling the use and training of these networks. You can see the comparison of the most used networks in picture 3.56 below. The accuracy of these networks was measured on ImageNet validation data using the same Graphics Processing Unit (GPU).

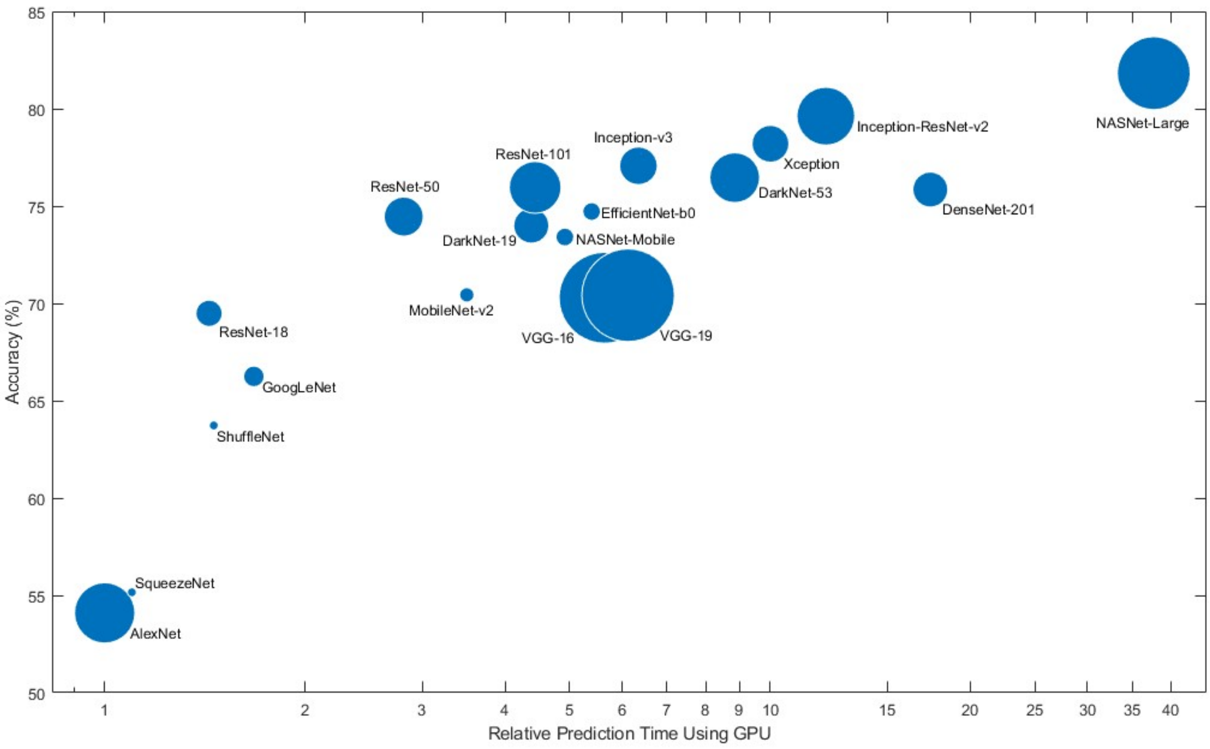


Figure 3.56: Comparison of open source CNNs [48]

We can see that the Inception-v3 [51] which was created by company Alphabet Inc. offers great accuracy to computational demand ratio. You can see the architecture of this network in picture 3.57 below.

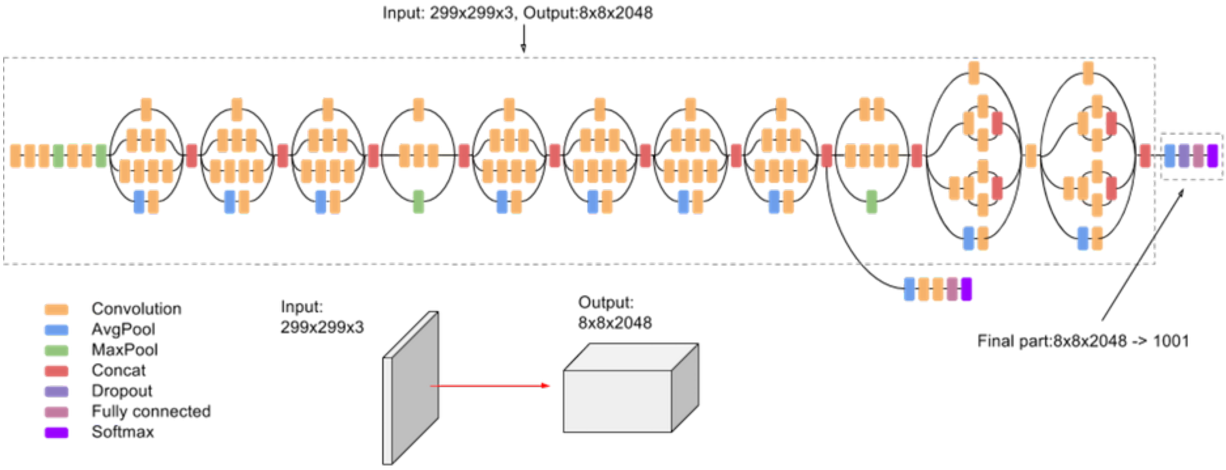


Figure 3.57: Inception-v3 architecture [51]

When working with neural networks there are multiple frameworks available. Let us show some of them.

Probably the most used one is TensorFlow [52]. It is an open-source software library for working with neural networks. It may be utilised with programming languages like Python, which is the most frequent one, Java or C++. It can be used along with Keras, which is a deep learning Application Programming Interface (API) enabling downloading pre-trained networks and editing them. It is handy when working with NN in Python. TensorFlow also has great features for creating a dataset. It can generate an adequate object with labelled data, shuffle, normalise, resize and divide into batches. This is very easy and saves time.

Another well-known open-source framework is Pytorch [54], which was developed by Meta Platforms in 2018 and can be used with Python, Java and C++. Some interesting deep learning projects are built using Pytorch, such as Tesla Autopilot or Uber's Pyro.

A similar product to the Pytorch that was released by Meta Platforms as well, is Caffe2 [55]. It is a successor to the Caffe, which was created at the University of California, Berkeley. The main difference between Caffe2 and Pytorch is that Caffe2 is built to excel at utilising both multiple GPUs on a single-host and multiple hosts with GPUs, so it is better for large scale deployments. It supports Python, C++ and Matrix Laboratory (MatLab).

The next possibility is to use MatLab along with its Deep Learning Toolbox [56]. However, this product is not free to use. It offers Graphical User Interface (GUI) that is very transparent. There is a possibility of downloading models from TensorFlow, Pytorch or Caffe2. The main advantage of using MatLab is that it is very visual and you can see the objects in the workspace. Additionally, you do not have to deal with incompatibilities and are able to create complex software for industrial applications where NN is just a part of it. The big downside is that one needs to buy a number of toolboxes to unlock additional features. For example, there is a requirement to own a Parallel Computing Toolbox in order to train using GPU.

In the end it is insignificant which framework is used for the vast majority of applications. What is really crucial is the correct architecture and most importantly a high-quality dataset.

4 Results and progress

The progress and results of the project are described in this chapter. It starts with a section covering the development of an optical cell, beginning with the concept and continuing with the design of the cell. The next section describes a machine vision system. First, the inspected parts are analysed. After that, the camera and lighting setup is discussed as well as the functioning of the defect detection software. The following section deals with the design of a robotic workplace, starting with the analysis of the two degrees of freedom (2-DOF) robot design, followed by a description of the two modes concept and the functioning of the workplace. The last section is about testing and provides accuracy statistics.

Each section describes a single part. These parts compose a system and are dependent on each other, which is depicted in the picture 4.1.

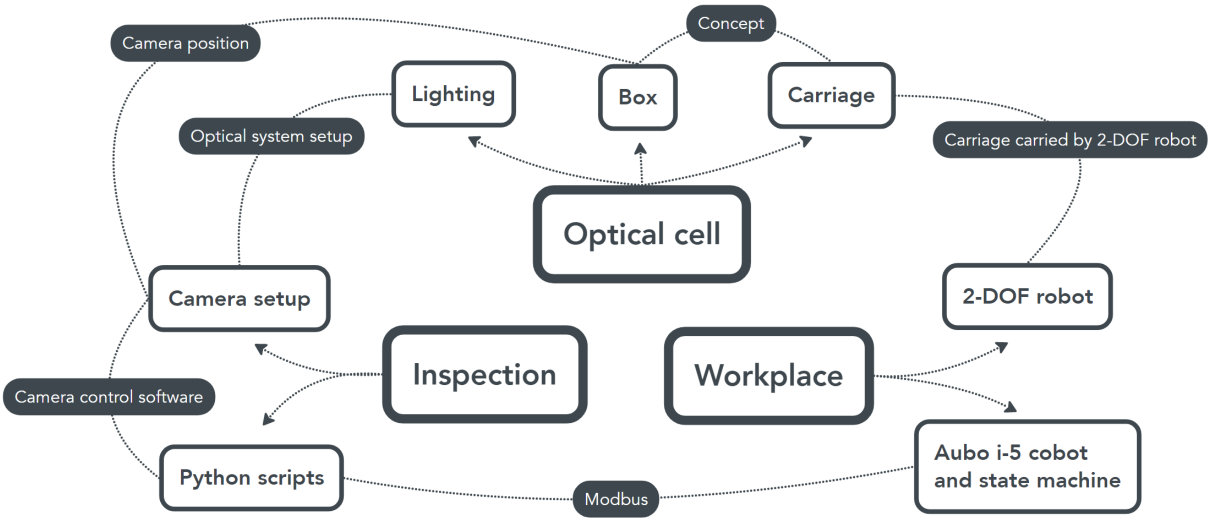


Figure 4.1: Mind map of project parts dependency

4.1 Development of optical cell

This section deals with the development of the optical cell. It is divided into subsections explaining the concept and describing the design of the final form.

4.1.1 Concept

The initial idea was that an optical cell would be developed that would allow various lighting conditions to be set and the position of the camera adjusted to create an environment to test machine vision applications. This unit would be part of a configurable robotic workplace and the inspected parts would be manipulated using a collaborative robot. It would not be designed for particular parts. Instead, it would serve as a tool for testing when developing an application for a potential customer and for trade fair purposes. This meant that the optical cell had to be as universal as possible.

At first, a few crucial decisions had to be made, starting with the question whether the concept of the box should be open, meaning that the inspection does not take place behind the complete cover, or closed to prevent the environment light from coming through. After thorough consideration, the closed concept was chosen because it gives certainty of constant light conditions, which is important because the workplace would be moved around the trade fairs where the light conditions are unpredictable. Another decisive factor was the fact that it allows the sides of the box to attach devices and protects the camera during transport.

It was established that there would be a clamping mechanism and that the robot would place the parts in it. Another thing that needed to be decided was how to scan as big a part of the object as possible in the shortest time. There were two ways to do it. Multiple cameras could be used, each positioned from different angles, or the part could be moved. The main advantage of having multiple cameras is that it saves time because it takes a while to move the part, whereas with multiple cameras, all the pictures may be taken simultaneously. However, as already mentioned, this application is mainly for demo and testing purposes, so the cycle time is not vital. On top of that, there was a lack of cameras on the market during the project development, and using each camera to take only a single image during a cycle is a bit of waste. It was hence decided that the part would be rotated and a single camera would take multiple images of it.

Next, it was necessary to find a way to place the part in the clamping system. There were two approaches considered. The first was that the robot would place the part directly in the clamping system, which would be fixed in the unit. The second was that the clamping system would be ejected from the unit to make the necessary space.

The closed concept is advantageous because of the constant light. But it also means less manoeuvring space for the robot. Additionally, there was quite a limited space for the unit in the workplace, so the camera could not be far from the object. These facts added up to the problem of space for the cobot. There were some ideas, such as opening the top of the box to make a way for the cobot, but it was immediately rejected, as there would always be a problem of potential collision with either the walls of the unit or the camera. The idea of placing parts in the unit directly by a cobot was thus not feasible.

That is why it was decided that the clamping system would eject. Still, there were many ways how to do it. It was clear that it would essentially form some kind of carriage that would move in and out of the box. The first idea was that it would be moved using an actuator, which meant that an automatic door had to be developed to keep the closed concept and make way for the carriage. For that reason, another linear actuator had to be added. This way, the concept of an elevating door was made. This seemed like a sensible solution, but when the design process started along with the search for concrete parts, a lot of problems emerged. An early stage model can be seen in the picture 4.2.

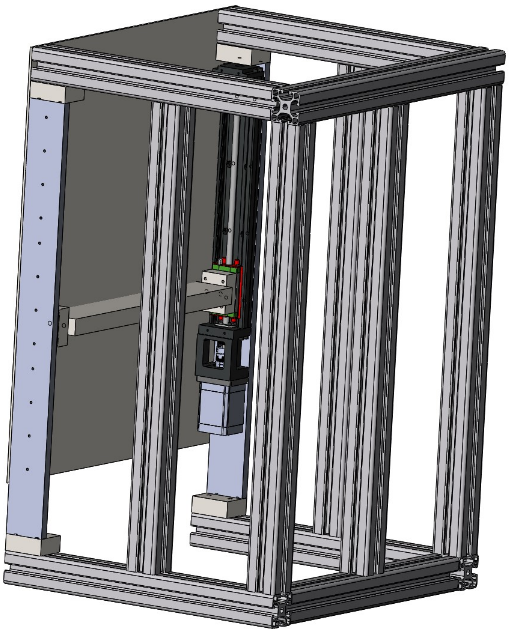


Figure 4.2: Early stage concept with the elevating door

The main problem was that the actuators with a sufficient payload capacity and stroke were too large and that the door system was not practical. Another reason for not taking this path was the fact that a 2-DOF robot, described in subsection 4.3.1, had already been planned to be part of the workplace. Going this way would mean not using it and adding two additional axes to the workplace. In the light of these problems and the fact that it would be redundant to add other axes, this concept was rejected.

Another approach was thus chosen. Instead of adding axes to the workplace, the 2-DOF robot would be used to carry the carriage. To preserve the closed design and not have to deal with the opening of the door, the concept of a box with an open passage was created. To close the gap and prevent the light from coming in, the carriage would be lifted by the 2-DOF robot. This idea was final and can be seen in the picture 4.3 below. To give access to the unit for the user to set up the camera, a manual door is used.

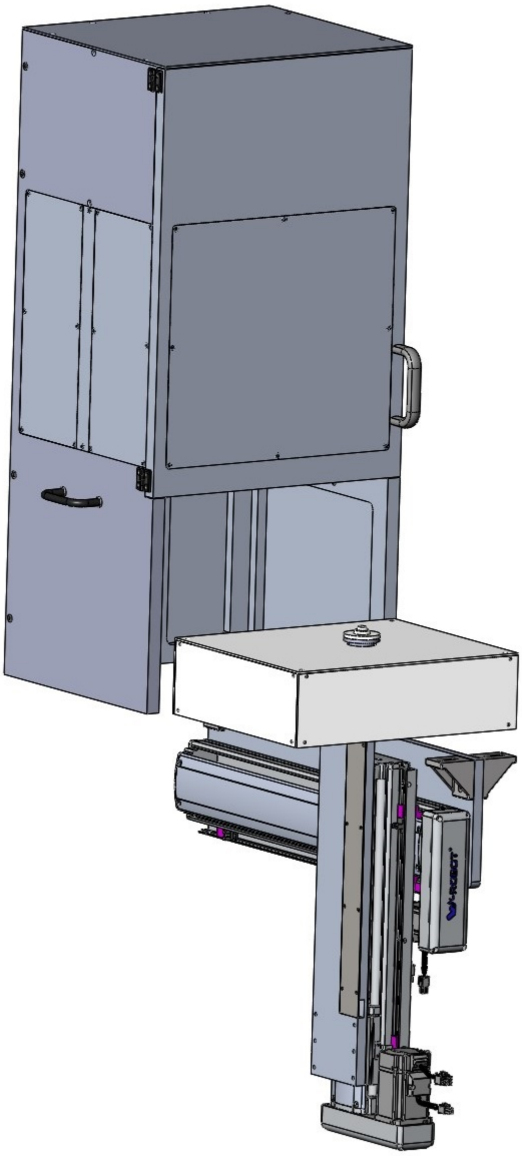


Figure 4.3: Final concept with carriage carried by 2-DOF robot

The axes, which can be seen in the picture 4.3, form the 2-DOF robot and are hidden inside the work table, so they do not take up any space outside. It also allows the regulation of the distance between the camera and the object not only by changing the position of the camera but also by changing the height that the carriage goes up to. The downside of this concept is that the box needs to be taller so that it makes space for the carriage to pass through. Still, it is better than to have a door that goes up and makes the box twice as tall.

The carriage contains a clamping mechanism for holding the part which is placed there using a collaborative robot. A system that rotates the clamping mechanism during the image capture is inside the carriage.

Inside the cell, there is an attachment system to position the camera freely and to hold additional lights. However, the lighting conditions should be adjusted primarily using the integrated wall lights. The idea behind this was that there would be shining walls producing dim light, regulated separately, with light conditions suitable for a given application. The design details of the whole system are discussed in the upcoming subsection 4.1.2.

4.1.2 Design

This subsection presents details on the optical cell design and is divided into subsections focusing on the box, carriage and lighting. SolidWorks software was used as a tool for 3D modelling and drawing.

Box

The final design of the box can be seen in the picture 4.3 but let us dive deeper into why it looks like this and what it features. After the concept was thought out, the proportion had to be established. There were two factors that played a key role in the decision-making.

First, there was already an existing worktable, which can be seen in picture 4.30 and its tabletop was of a limited size. The distribution of the workplace is discussed in subsection 4.3.2 but for now, let us work with the information that we had an area of 500x400 mm for the box itself, which was not too much.

The second factor was that there had to be enough volume for the camera to be variously positioned, to fit in the box, and maintain an optimal working distance. It was decided that the maximum size of the inspected object would be a cube with each side measuring 100 mm as this seemed like a proper value for the intended purpose.

As mentioned, the box featured a gap for the carriage to pass through. The size of it was based on the height of the carriage, which is 110 mm, the height of the clamping system, which can have various heights but probably not more than 50 mm, and the object, which has the maximum height of 100 mm. There also had to be a reserve for the object to go under the door, and the carriage had to be slightly above the work table. That is why the gap for the passage was set to 300 mm.

The carriage was planned to go up so that its surface is at least 50 mm above the gap in order to properly block the incoming light but most importantly the wall lights begin at that height. After that, there still had to be enough room for the camera setup, which is approximately 150 mm long, for the inspected part, as well as the clamping mechanism, and there also had to be optimal working distance maintained, which is approximately 150 mm. Therefore, the final box is 900 mm tall.

The base of the box has inner dimensions of 310x370 mm. There had to be enough space between the object and the sides of the box so that if the camera was pointed towards the object from an angle, the sides, which might contain other devices, would not be visible. The reason for it being rectangular is that there is an option for the camera to be attached from the side in order to inspect the profile of the part. One side had thus to be wide enough to create an adequate distance between the camera and the object.

After the dimensions were established, the design process began. The first idea was to produce the box using aluminium profiles because they are generally delivered quickly and the assembly is easier to reconfigure thanks to them. However, it turned out to be impractical during the design process because to attach additional components such as door hinges, or to make space for the wires, additional machined components would need to be manufactured. This would lead to the situation where the profiles would serve only as a frame and multiple components would have to be added only to make the attachment of certain parts possible. Therefore, this concept lost its purpose, and it was decided that the box would be created using aluminium plates that would be designed so that everything would fit and no additional interstage components would be necessary.

As was already mentioned, the walls of the box would feature integrated lights. The specifics of this lighting will yet be discussed, but the aluminium plates had to be designed accordingly. The idea was that each wall would be formed using multiple layers and create a "sandwich"concept. The inner side of the box would be made up of a transparent and nontransparent type of 3 mm thick plexiglass. Thus, the plates had to have space for these plastic boards, as well as thread holes for them to be screwed together.

It was determined that the lighting would be fitted to each wall, including the doors, and would make a 320 mm high zone. In this area of lighting, there were holes designed to fit the LED lights covered with a transparent plexiglass on the inner side and a 3mm thick aluminium sheet holding the lights on the outer side. This way, the lighting could be accessed from outside the box in case of replacement. In order to produce monolithic light, the LEDs had to be in a certain distance from the plexiglass. Generally, the further the better, but very thick aluminium plates would be heavy and expensive and would enlarge the whole box. Therefore, after some tests it was decided to have the walls made of 20 mm thick aluminium plates to make an approximately 12 mm gap between LEDs and plexiglass. The 3D model of the aluminium plate used for the back side of the box can be seen in the picture 4.4.

The lights needed to be powered by wires. For the purpose of creating a path for them, wire canals were designed that lead through the aluminium plates and exit the box at the bottom. The wires were then led under the work table. There was also a canal for the cables that power the camera added. The wires in these canals are covered using non-transparent plexiglass and can be easily accessed. The canals can be seen in the picture 4.4 of the 3D model and the reality of cable management can be seen in the picture 4.5.

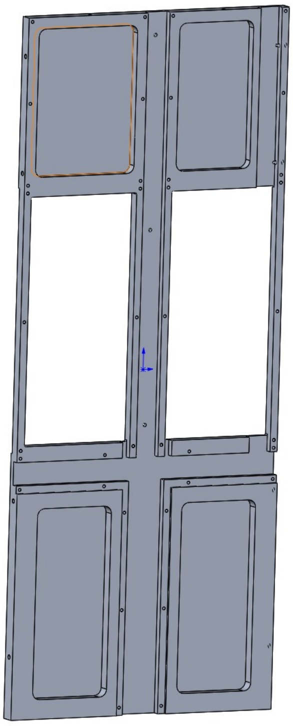


Figure 4.4: 3D model of back side of the box

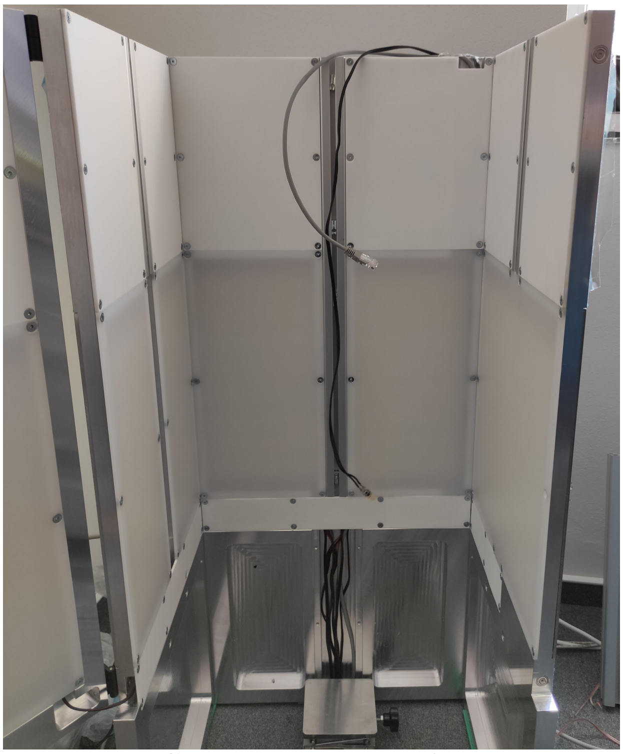


Figure 4.5: Cable management of the real box

A manual door was developed to allow access to the camera. It turns on two hinges. To prevent the door from opening arbitrarily, two magnets were used. For comfortable opening, a handle was added along with two others on each side of the box for easier carrying. The top of the box was covered with a 3 mm thick aluminium sheet.

To allow the user to position the camera, an adjustable camera bracket was developed. It consists of a square aluminium profile frame that has one additional profile in the middle. This profile can be moved inside the frame, enabling linear movement in one direction, and has a ball joint attached. This ball joint enables a linear move in another direction as well as any rotational movement and has a camera holder attached to the end.

In order to attach this whole system to the box, there were flat aluminium profiles designed to fit in the middle of each wall except for the doors. Using these profiles and steel angle brackets, the frame is allowed to make a linear movement up or down to set the distance between the object and the camera. These profiles serve not only for the attachment of the camera bracket but also for additional lights. This solution can be seen in the picture 4.6 below.

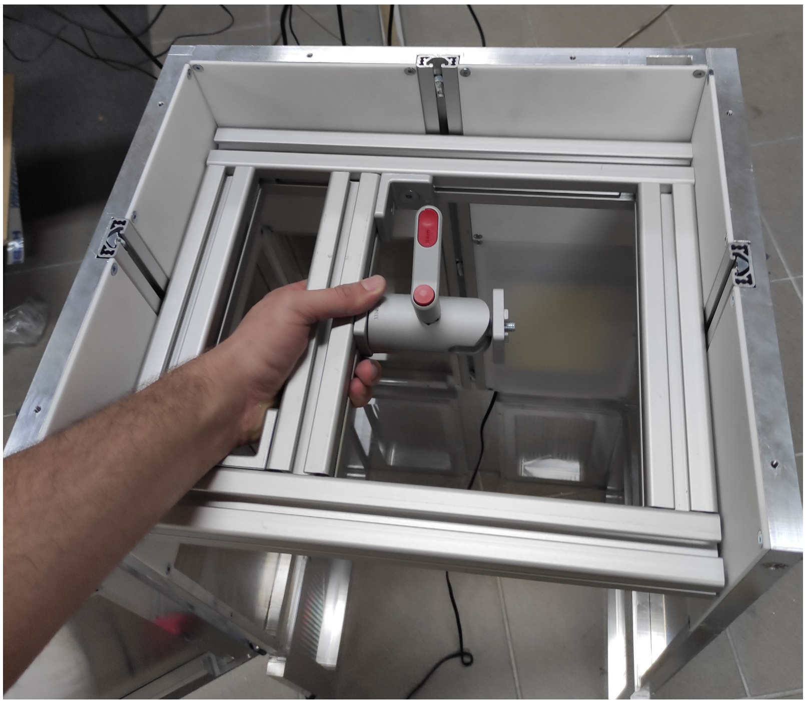


Figure 4.6: Camera positioning mechanism

There was also an optional system designed to attach the camera from the side. The profile on the left-hand side of the box was split into halves, so that one half could be put away. This way a gap for the camera was created. The side attachment system consisted of bent steel sheets and can be seen in the picture 4.7. The final form of the whole box can be seen in the picture 4.8 below.

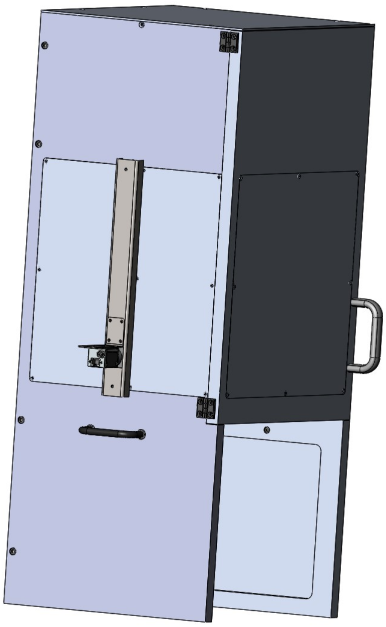


Figure 4.7: Side system for the camera

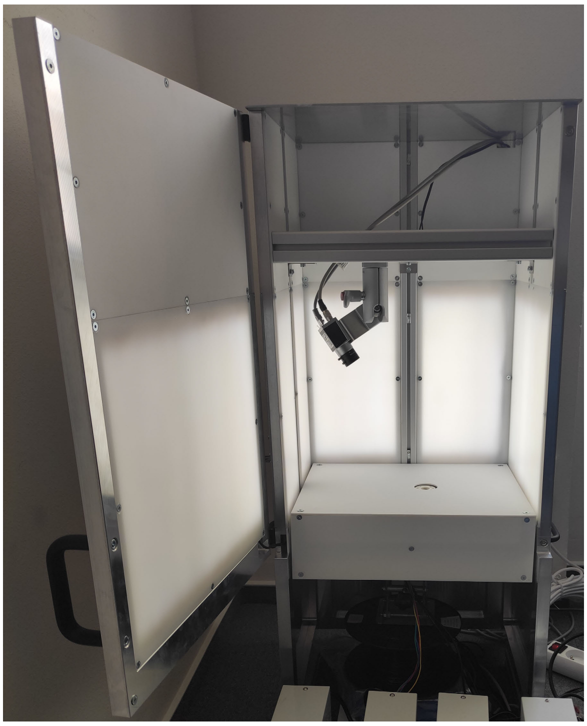


Figure 4.8: Final form of the box

Carriage

Another vital part of the entire optical cell is the carriage. This device serves multiple purposes. The first one is that it creates a certain shutter preventing the light from the environment from coming to the box, the second one is that it carries a rotational system with the clamping mechanism. Furthermore, it contains a device that produces electrical pulses to trigger image capture in the camera. The final design can be seen in the pictures 4.3 and 4.8, but again, let us discuss it in more detail.

One of the main requirements of the carriage was to be as low as possible because the height of the box depended on it. Therefore, it was vital to make the rotational system as low as possible. It was clear that an electric motor would have to be used as a powertrain. It would not be ideal to place the clamping mechanism directly on the electric motor as the maximum value of the weight of the inspected part was decided to be 5 kg and it would axially overload the motor. For that reason, an output shaft with the proper bearing would have to be added, as well as a clutch to connect it to the electric motor.

Clearly, the output shaft had to aim in the direction of the height of the carriage. That is why putting the components in the line was not an option and they had to be split into more directions in order to create a low carriage. To achieve that, a gear or belt system had to be added. At first, there was a concept of two shafts, one propelled and one propelling, that would be connected using a cogged belt. However, this idea turned out to be impractical, as both of these shafts had to be mounted with bearings. That is why an angular transmission with 1:1 gear ratio was used. The precise model is Mädler KEK 41000103. It is very compact, only 42x42x24 mm, and allows 2.5 Nm of torque and 120 N of axial force to be applied.

The stepper motor was decided to be used because it is easy to operate and offers a great torque to size ratio. It was decided that one full rotation should take approximately two seconds. The moment of inertia of the 5 kg 100x100x100 mm cube is 0.0083 $kg.m^2$. Using the combination of Newton's second law and the relation between acceleration, time and position, we get the equation 4.1:

$$M = \frac{2.\varphi.I}{t^2} \tag{4.1}$$

, where φ is the angular position, I is the moment of inertia, t is the time and M is the momentum. If we substitute φ for π representing half of the rotation in radians, I for the moment of inertia of the mentioned cube and t for 1 s, we get the correct value of the momentum as it has to accelerate and decelerate during a single two seconds turn. The final value of M was 0.052 Nm, which was very low and meant that we could choose almost any stepper motor. At first, the smallest possible stepper motor National Electrical Manufacturers Association (NEMA) 17 was considered, but these models generally have small shafts, and in the light of the fact that the transmission has a shaft of 10 mm in diameter, the difference would be too big. Thus, NEMA 23 with an output shaft of 8 mm in diameter was used. This motor is excessive, but at least it can use the full potential of the transmission and allows an almost instant reach of the desired angular velocity. The stepper motor was fixed using a Pololu NEMA 23 holder. A flexible clutch was used to connect the transmission to the motor to compensate for the imprecise alignment of the shafts.

There was a variable fast changeable clamping system at the end of the transmission output shaft. It consisted of a flange fixed to the shaft and a clamping mandrel, which would be designed for each inspected part separately. The whole rotational mechanism with the 3D printed flange can be seen in the picture 4.9

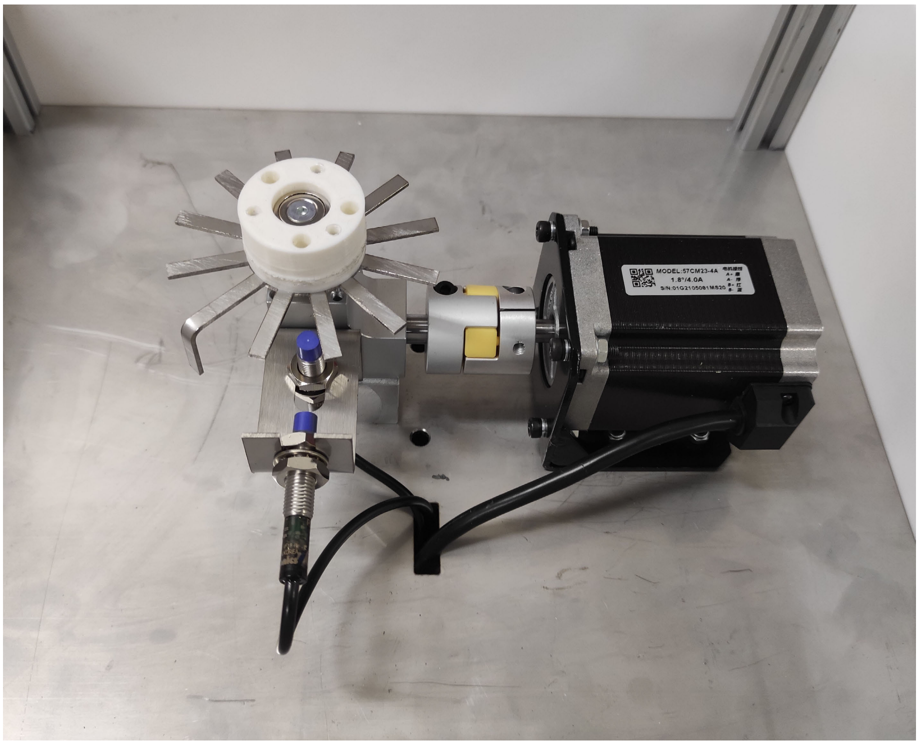


Figure 4.9: Rotational system

The idea behind the fast changeable system was that the flange would have three holes where the pins of the clamping mandrel would be inserted. This way, the mandrel would be fixed to the flange. The holes were arranged asymmetrically to ensure that the mandrel could be inserted only one way. Furthermore, the flange was screwed to the output shaft of the transmission to be axially secured, and had a magnet in the centre, which prevented the clamping mandrel from ejecting. Three different clamping mandrels designs can be seen in the picture 4.10, each serving for different inspected parts. The bottom part of the mandrel is the same at each design but the top changes in order to fit for the required application.

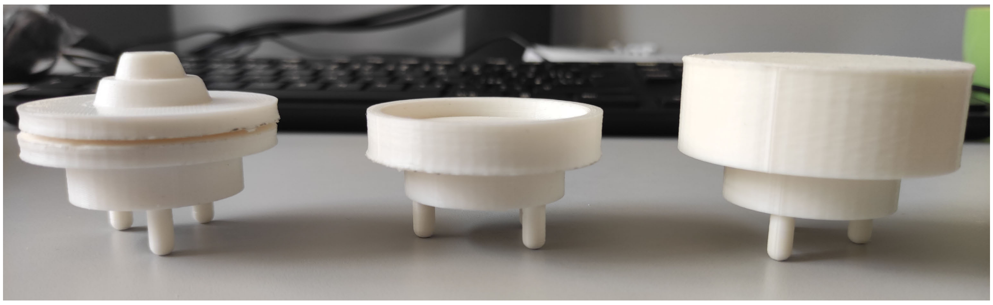


Figure 4.10: Clamping mandrels

To precisely and evenly capture the images during the rotation as well as to home the rotational system, a triggering mechanism was developed. It can be seen in the picture 4.9 and consists of two inductive sensors and a coding disc, which was designed to have 12 arms for camera triggering, where one of them was longer and bent for homing purpose. There was also a holder for the sensors developed, allowing the user to set up the position of them. The coding disc was screwed to the flange. The details of how it works with the camera are discussed in the subsection 4.2.2.

The whole mechanism described above was designed to lie on a 3 mm steel sheet, which would be attached to the 2-DOF robot, as can be seen in the picture 4.11. The sides and top of the carriage consist of the same non-transparent plexiglass as the one used in the box. They were attached using aluminium profiles, which were screwed to the steel sheet and created a frame. The carriage with all the plexiglass boards can be seen in the picture 4.8.

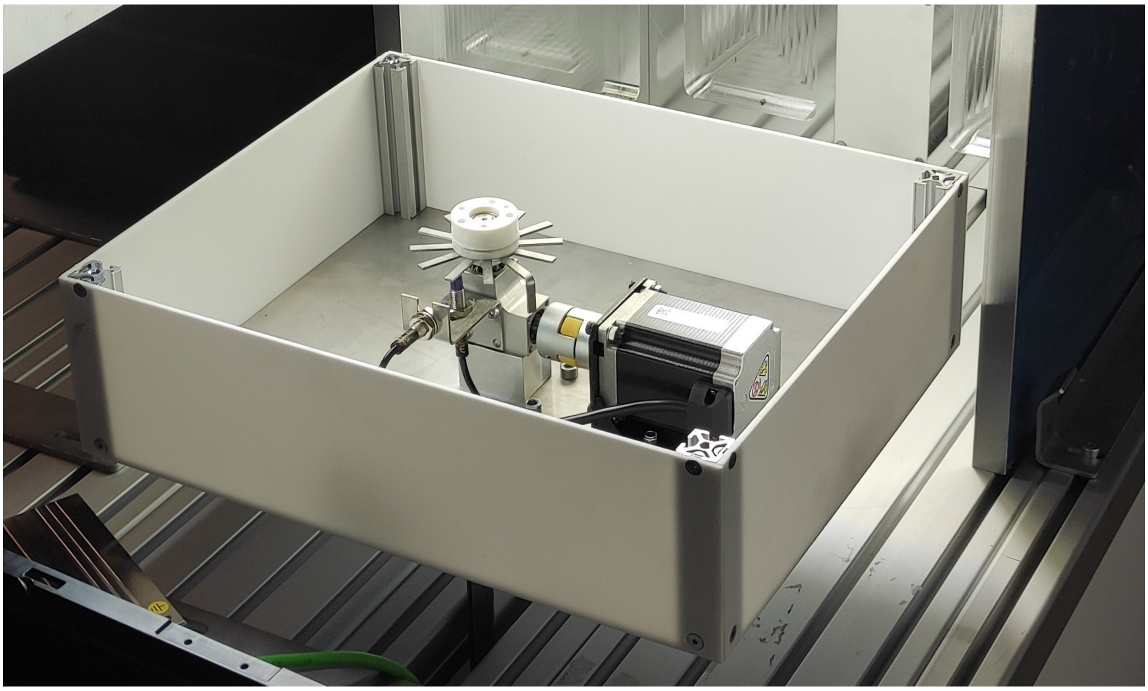


Figure 4.11: Carriage attached to the robot entering the box

Lighting

The integrated lighting was mentioned several times but never discussed, so let us provide some details. The optical cell should be as universal as possible. Therefore, the lighting conditions have to be adjustable. The specific types of lighting were analysed in the subsection 3.2.3 of the theoretical survey. Correct lighting should in general create a situation where the defects are in contrast with the rest of the scene, and no interfering elements are visible.

This is obviously very difficult to accomplish with no knowledge of the inspected part. That is why the idea was to make lighting that could be adjusted according to the situation and to design the inner surface of the box as clean as possible to minimise the amount of devices that could create a reflection. The initial thought was that there would be LED panels in the walls covered behind the diffusive plexiglass boards, producing a dim evenly distributed light that would eliminate shadows. For the purpose of setting up specific lighting conditions to highlight the defects, additional lights would have to be attached to the box.

After some thinking, the idea to make the integrated lighting more powerful and adjustable took over. This way the final concept was created. To achieve this, it was essential to make room for the lights in the walls, which was already discussed, and to choose the right light source and power source. Furthermore, it was necessary to find the correct material for the diffusive boards.

It was clear that the lighting would not be all over the walls as it would be impractical and useless in the camera area. Also, keeping the shiny aluminium inner surface would not be great, as it could create reflections. There would thus be two types of plexiglass, one covering the lights and one covering the walls in the area where the lights are not present. Each type required different material parameters.

The interior of the box as well as the carriage was to be white, as this makes a bright environment where the light reflects. The white colour also creates a sharp contrast between the environment and the inspected part, so it is easier to perform segmentation if necessary. The plexiglass used to cover the aluminium plates had to be white and as matt as possible to prevent reflections. The specific model SATINGLAS 54000 from Omniplast s.r.o. was chosen after a personal visit of the company's showroom.

The material used to cover the lights was quite difficult to select. The most important requirement was that it had to evenly distribute the light, so that the whole lighting segment would appear monolithic. A bar light made by the professional manufacturer Wordop was analysed and it was found that monolithic light was achieved mainly using small plastic lenses covering the LEDs, while the plexiglass was almost transparent and did not have much effect. This approach proved great, as all the light potential is used and no energy is lost in the diffusive layer, but it could not be used in this prototype.

For that reason, another approach was taken, which uses a highly diffusive board even at the cost of the light potential being sacrificed. As mentioned above, the final model of the plexiglass was chosen after a personal inspection and testing at the showroom of Omniplast s.r.o. The specific series was SETA-LED GREEN CAST 17001 with light permeability of 53%. It offers decent diffusive qualities but is rather glossy and reflective. This would not be a problem if the lights were on all the time, but the assumption was not all the lights would be during some applications.

The plexiglass boards were hence covered with a transparent matt film. This solved the problem with reflections and it multiplied the diffusive effect. The board without the matt film can be seen in the picture 4.12 and after applying the film in the picture 4.13. All boards were cut using the water stream cutting technology.

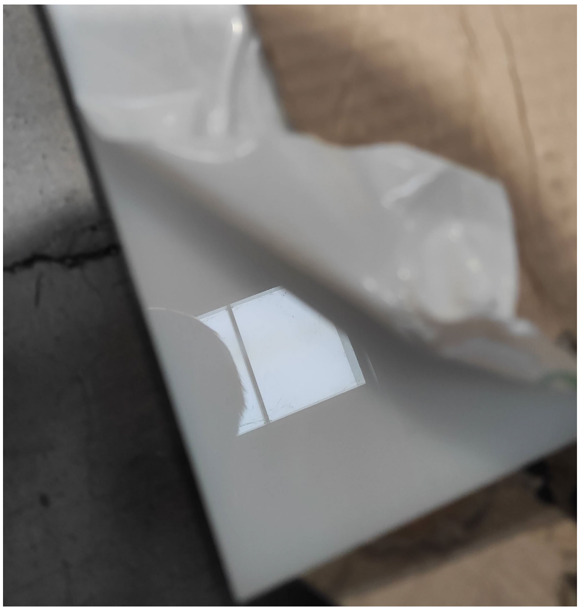


Figure 4.12: Plexiglass without a matt film

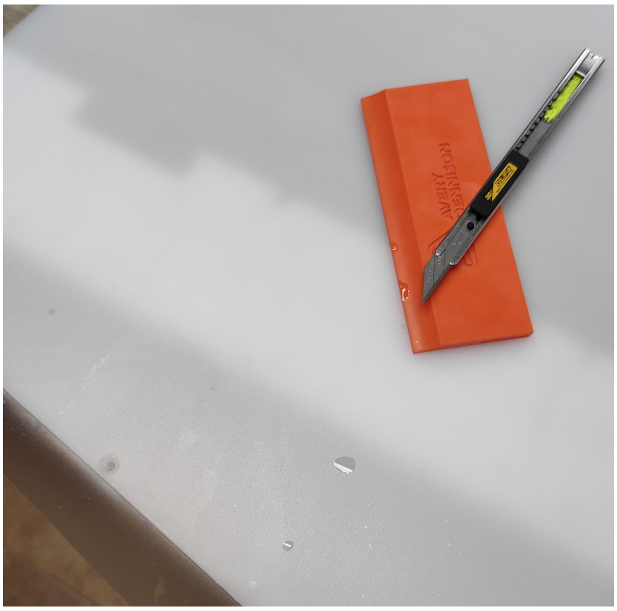


Figure 4.13: Plexiglass with a matt film

After the boards were chosen, it was necessary to choose the correct light source. The first idea was to use Chip On Board (COB) LED panels as they generally do not have the light points visible as much as average LED panels and therefore produce an areal light plane. There are also various sizes, with some large variants available. This way the required areas on the walls could be easily filled. After some calculations, specific models were ordered and placed inside the wall, as can be seen in the picture 4.14.

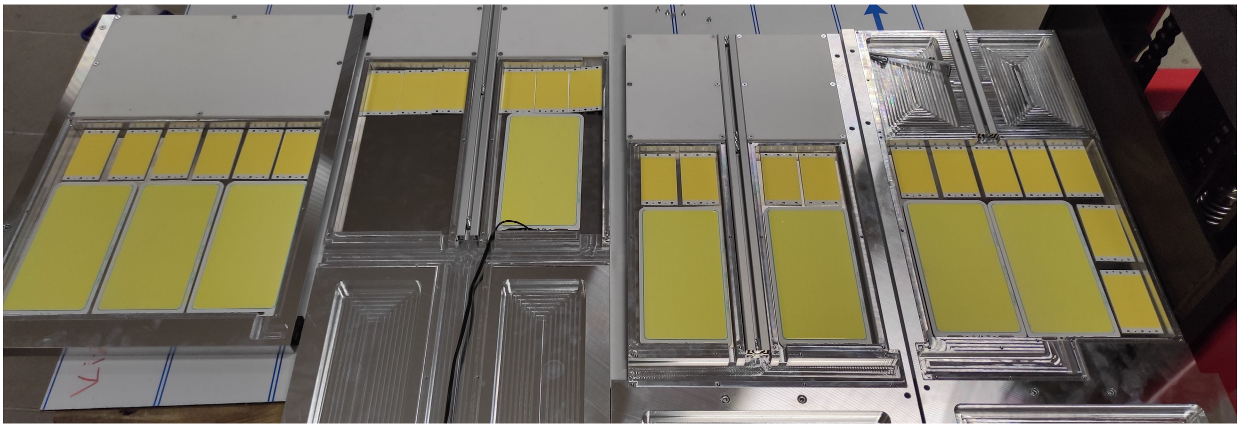


Figure 4.14: COB panels distribution inside the walls

The panels were then parallelly connected to the power source and tested. At first, they appeared very good. This setup had a very high lighting power and could be regulated by changing the voltage. But after a closer look there were a few problems. The first was that different panel models started to shine at different voltages even though all panels were made by the same technology. For example, at 8 V some panels were already shining while others were still off, which, of course, produced a nonuniform light.

Another problem was that each model had different light temperatures, which led to creating patches. After some tests, it was thus decided to switch to COB LED strips instead of COB panels. The 336 diodes per metre of LED strip density was chosen because it offered satisfactory light power and was not too dense. The model with a light temperature of 6500 K was selected, because it is generally better to have a colder light in industrial applications. These LED strips can be cut into segments. Joining these segments together makes a parallel connection. Therefore, the strips were arranged into lines with a 10 mm distance and soldered together. This was incredibly tedious work as there were 36 metres of a LED strip used which were cut into approximately 120 LED strips. This is shown in the picture 4.15. The final result meant a great improvement. This concept really produced a uniform light as you can see in the picture 4.8 and turned out much better than the COB panels, as can be clearly seen in the picture 4.16.

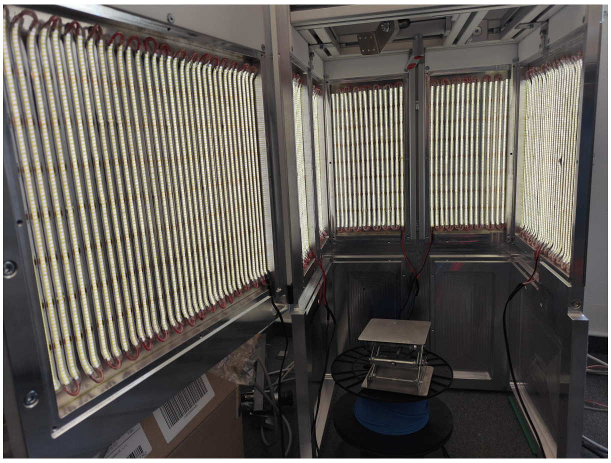


Figure 4.15: LED strips concept

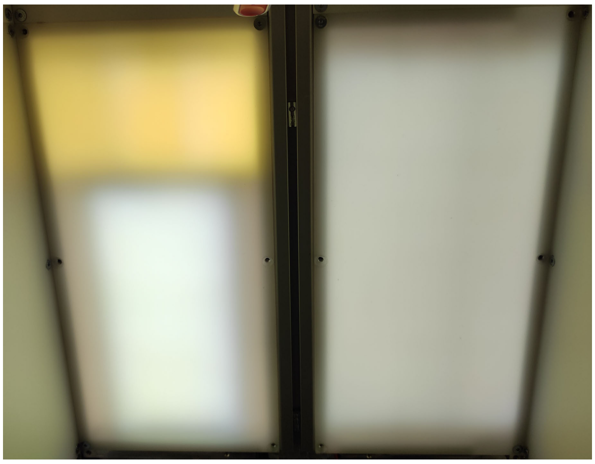


Figure 4.16: Panels and strips comparison

It was decided that each wall would have its own power source to be regulated independently. It was measured that with the 13 V voltage, approximately 3.5 A of current flowed into the largest wall, i.e. the door. That is why a power source of at least 50 W was necessary. Furthermore, it needed to be adjustable in the range of 7 V to 13 V because that was the range where the light intensity changed. If the voltage was lower than 7 V, the light was off, and voltages above 13 V could break it. The source had to be stabilised to prevent a blinking light, which would interfere with the camera. All requirements combined meant an optimal industrial power source could not be found.

As a result, laboratory power sources were opted for. After some testing, it appeared to be a good decision as it met the requirements. In addition, they have a display showing the actual voltage. This way, it was easy to return to the same conditions after changes. For each application, the notes about voltage for each wall would be simply written down, so the conditions could be easily reproduced if needed.

4.2 Optical inspection

This section gives insight into optical inspection. At first, the inspected parts used for testing are described. Next, the optical system that was used is discussed, including the software for image capture. Lastly, the software used for defect detection is described, covering the used dataset, architecture, and training progress.

4.2.1 Inspected parts

One of this project's objectives was to develop software for detection of defects and test it on various parts. Two objects with different appearance and types of defects were selected for this purpose. The decision to use these two specific parts was most importantly based on available samples, as it is generally difficult to obtain a large number of sorted samples.

The first and main inspected object is a damper used in the automotive industry. There were two types of defect to be inspected. The first type was the lack of filling on the bottom skirt of the damper. It can be seen in the picture 4.17, where the faulty part is on the left and the correct one on the right. The second type of defect was a missing part of the upper skirt of the damper. The comparison between the correct and the faulty part is in the picture 4.18, where the defected part is on the right and the correct one on the left. There were 50 samples of correct parts, 50 samples of defected parts of the first type, and 50 additional samples of defected parts of the second type. The shape of defects did not vary too much in either category. The main differences were in the size. This was not surprising as these defects were probably caused by some specific problem during manufacturing.

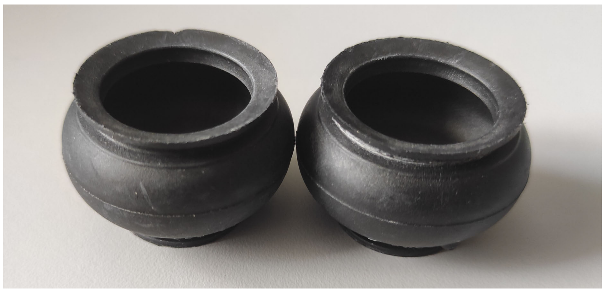


Figure 4.17: First type of defect of damper

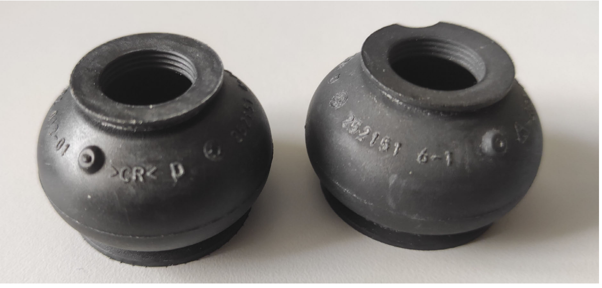


Figure 4.18: Second type of defect of damper

Fortunately, these parts were matt and it was clear that the reflections would not be an issue. A potential problem could be the fact that these parts drastically change colour of the surface from black to pale grey once deformed. This is probably caused by a thin film that is on the surface of these dampers.

The second object selected for inspection is a metal part of unknown function. It is symmetrical and has a glossy surface. There was only one type of defect to be detected. The task was to detect whether the surface treatment was applied correctly. The difference between the correct part and the defected one was in colour. Sometimes the faulty ones also had marks on the surface. The comparison between the correct (on the left) and the defected part (on the right) can be seen in the picture 4.19. There were 10 samples of the defected parts and 10 samples of the correct ones available.

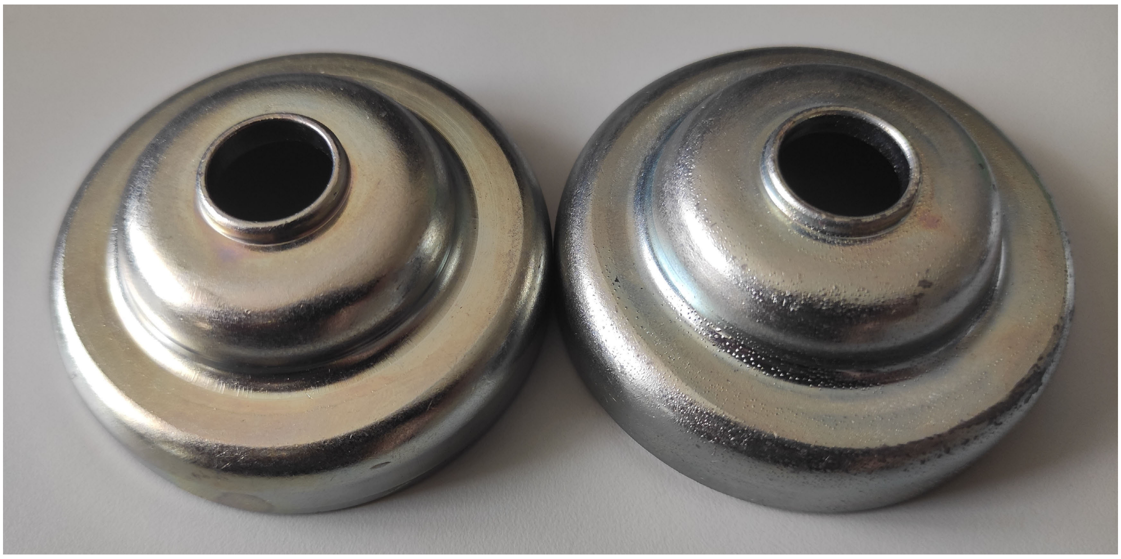


Figure 4.19: Comparison between correct and faulty metal parts

Originally, there was a plan to detect defects in glass coffee cups. But it was abandoned because the client provided a very limited number of samples. Only one sample representing each defect, to be exact. Because of that, there was no chance of creating a generalised neural network. Also, due to the lack of samples, there was no way to test it. There was a specific clamping mandrel designed for each part, all of them can be seen in the picture 4.10.

4.2.2 Optical system and lighting setup

The design of the optical cell and lighting was discussed, but the optical system used for the inspection was not yet mentioned. Therefore, let us dive deeper into the camera setup and how image capture works.

It was clear that the industrial camera would be used as it is compact and designed for these types of operation. The products manufactured by the company Basler AG had been used for many applications in the company Kinalisoft s.r.o. and there had never been any problems. This is why this brand was chosen. Still, there were many models to select from, each with different parameters.

It was already established that the pictures would be taken during the rotation without stopping. Since fast moving objects are distorted when a rolling shutter camera is used, it was essential to choose a camera with global shutter. This was described in greater detail in subsection 3.2.1 of the theoretical survey.

The next important factor was the type of communication. Typically, there are two options to choose from, USB or Ethernet. Ethernet is very handy as the camera can be simply connected to a switch and as long as it is on the same network, it is possible to get connected. It is nonetheless necessary to have a POE switch to power the camera. The USB is thus a good choice when it is planned to stay connected directly to Personal Computer (PC). In this case, it was opted for a camera with an Ethernet port.

Another key parameter was the resolution. Applications for defect detection do not normally require a big resolution. More resolution means more information, which sounds like an advantage, but it drastically increases the memory and computing power requirements during the CNN training, as well as during prediction. On top of that, it transfers a massive amount of data when capturing high-resolution images at higher FPS and could lead to a situation where the communication bus is overloaded and some images may be lost.

On the other hand, a low-resolution camera is not ideal either because smaller defects could be blurred and not visible. This could lead to a situation where the defects represented by the pixels in the matrix would be deformed. When there is an option to choose between a high-resolution camera, meaning 5 MPx and more, and a very low-resolution camera, meaning Video Graphics Array (VGA), a high-resolution one is typically preferred because the images can always be resized in post-processing.

The best option is to use a camera with a sensible resolution. Apart from resolution, the format of the sensor and its size matter. In our case, it was decided to get a camera sensor with the number of horizontal pixels as close to the vertical ones as possible, in order to make a square image. Larger sensors are generally better, because they simply capture more light.

Another thing to choose was the FPS rate, which defines the maximum number of images that the camera is capable of capturing per second. In our application, it was not necessary to have a high-FPS camera because even though the object moved, it did not move extremely fast and capturing a large number of images in a short time was not required.

The last but very important parameter was whether the camera would be colour or greyscale. Greyscale is sometimes sufficient, as it is not so demanding on the computing power when used with neural networks. Furthermore, the high contrasting defects may be well visible. Yet, as the future applications were unknown and the optical cell was supposed to be universal, a colour camera got preference as it gives room for more universal use, including the colour inspection.

After considering all these parameters, the model acA2040-35gc was chosen. It features the Sony Pregius IMX265 sensor, with a resolution of 2048x1536 px, and an electric global shutter, which was described in subsection 3.2.1. This is a colour camera capable of the 35 FPS rate, which was sufficient. It uses Ethernet for communication. Importantly, this model was one of the few available, which was fortunate because there was a lack of cameras on the market at the time this project was developed, as was already mentioned.

The next step was to choose the right lens. A fixed focal length lens was opted for as it is universal, and most importantly, really compact. By contrast, a telecentric lens is large and would take too much space in the box and would make positioning difficult. The camera uses the C-mount for lens attachment. The optimal focal length for this application was computed to be around 10 mm using the Basler online calculator.

Kinalisoft s.r.o. is a distributor of Coolens lenses manufactured by Shenzhen Vico Technology Co., Ltd. For that reason, it was decided to choose a model from this company. After considering all available models, two lenses were chosen. The first was MFA1-118-5M8 with an 8 mm focal length and the second was MFA1-118-5M12 with a 12 mm focal length. The whole optical hardware setup can be seen in the picture 4.20 below.

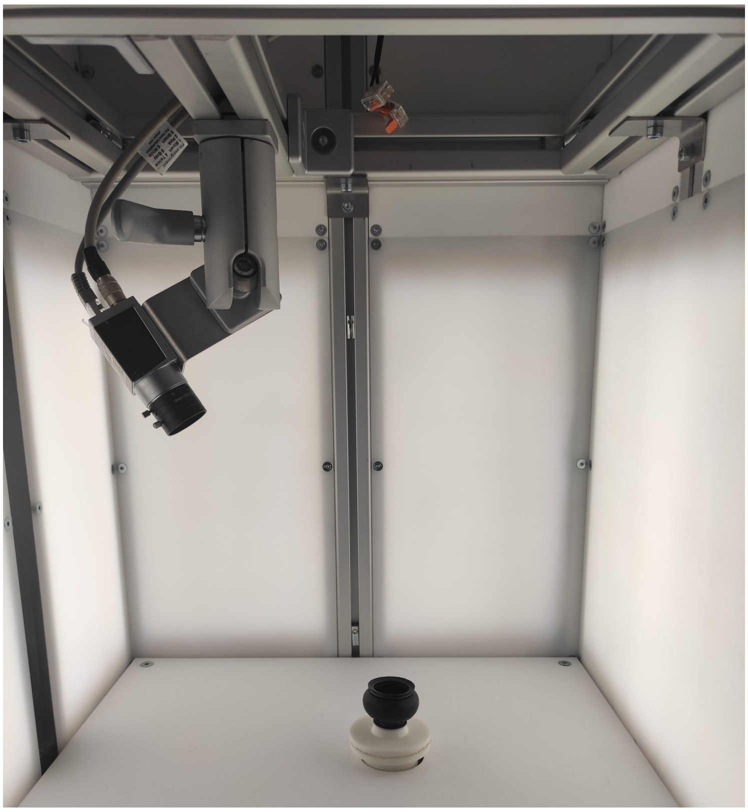


Figure 4.20: Optical system setup

Both of these lenses have magnification adjustment for focus, and the aperture adjustment, which controls the amount of the incoming light. In order to get the sharpest possible photo of the object, it was naturally necessary to be able to adjust the focus based on the working distance. When adjusting the aperture, the aim is to create a well-balanced high contrast photo. This means that when the aperture is opened too much, all objects in the image tend to lose their real colour and appear too bright. Conversely, when it is too closed, all objects appear too dark.

To produce a sharp high contrast image, it was necessary to balance the settings of the camera with the lens adjustment, since the sharpness and light balance of the image does not depend only on the lens, but also on the camera, most importantly, on the exposure time. That being so, let us discuss the camera setting.

The camera can be controlled using the Basler pylon Camera Software Suite, which is very easy to navigate through desktop application. It serves mainly for setup purposes. First, the camera needs to be powered and connected to the same network as the PC. Then the camera becomes visible and can get connected. Once the communication is set, the camera view is accessible. Also, a settings menu appears with a lot of categories. The image can be cropped, axially reversed, etc., but these operations can be done in post-processing.

The more important setting is the exposure time, which is a value in microseconds. The longer the exposure time, the longer the sensor is exposed to light, which in turn impacts the image brightness. The image brightness needs to be adjusted with both the exposure time and aperture. But there is more behind it. The more closed the aperture, the greater the depth of field is, meaning objects remain sharp in a wider range of distances. Hence it is usually best to have a longer exposure time with a smaller aperture. However, this has a downside to it. In order to produce a sharp image, the scene may not change during the exposure. It follows that when capturing moving objects, the exposure time has to be short otherwise the object is blurred.

Another important setting is the white balance. The camera has an integrated automatic white balance system that can be turned on or off. It was decided that it is better to have this feature off because even though the image looks better, the automatic white balance adjustment might differ over time, leading to inconsistency.

The next practical feature of the camera is that it has an image capture trigger. It can be either software or hardware. The camera has an additional 6-pin connector at the back of its body for additional control, and the hardware trigger is one of them. If the camera is set to the trigger mode, it captures a single image with each pulse received.

There are many more adjustable options, such as communication setting, etc., but these are typically left to default settings and are used only in specific cases, because the majority of these functions can be done in post-processing.

This application mainly helps a user to find the best camera position and to set correct settings. Besides that, Basler has a Python library called Pypylon, which allows the user to set up the camera and control it using a Python script. All the settings discussed here can also be adjusted using this library. There is also a GitHub page [57] where a lot of sample code is provided. It was utilised when the script for camera control was written.

Several Python scripts were produced during the development of this project. They serve for dataset creation, image editing, CNN training, and final testing. Those for dataset creation and for final testing both contain a part of the code that handles the camera. It works in cooperation with the triggering mechanism in the carriage, which was described in subsection 4.1.2.

The first thing done in the script is that all visible cameras on the same net are found. Then the one with the correct Identification (ID) is opened and its parameters are set, including the exposure time, etc. The image capture takes place in a never-ending loop, where the grabbing start is initiated. It stays in this loop as long as the script is running. The camera is set to take a picture using a hardware trigger, so in this loop a picture is taken only when the camera receives a pulse.

These pulses are generated by a Programmable Logic Controller (PLC), which is a part of the workplace and has other tasks as well. The details are given in section 4.3. In terms of image capture, the system uses one of the inductive sensors, shown in the picture 4.9, to generate pulses when the coding disc moves above it. This way the programme in PLC knows when to generate a pulse for the camera. The reason why the sensor output is not connected directly to the camera is that the pulse sent to the camera must be of a defined length.

To sum it up, there is a Python script and PLC. The Python script is used to initialise the camera and after that stays in a never-ending loop. Whenever the PLC detects an edge in the signal from the inductive sensor, it generates a pulse, and the camera takes a photo, which is loaded in the Python script and is further utilised

4.2.3 Inspection software

The optical cell was described in detail in the previous sections mainly from a hardware perspective. The main aim of this subsection is to describe how defect detection software works.

This software was supposed to use methods of artificial intelligence. Therefore, quality inspection was performed using a classification CNN, which was trained on the acquired dataset containing images of defected parts mentioned in subsection 4.2.1.

The whole software was written in the Python programming language. It was decided that the Tensorflow 2 would serve as a framework for creating the CNN because it is relatively easy to work with. Moreover, many examples can be found on the TensorFlow website [52]. Some of these examples inspired the development of the scripts and some parts of the code were used. There was also a Keras API used and the examples from its website [53] provided inspiration.

These scripts were run using a Docker container, which is close to a virtual computer running on the Linux operating system. This was not created by the author of this thesis but was provided by an Information Technology (IT) specialist from the company Kinalisoft s.r.o., because when training neural networks using Tensorflow and GPU on Windows, it is essential to have the right version of the GPU driver corresponding to the correct version of Tensorflow and Python. This may sometimes lead to problems. Especially when trying to run the script on a different PC.

By contrast, when using the Docker container, it is easier to have compatible versions, and there usually are no problems. Furthermore, when trying to run the scripts on another PC, the whole container can be transferred, with all the libraries and drivers. There is hence no need to install Tensorflow, GPU drivers, etc. again, since all this is already packed in the container.

It was really easy to use, as all the script editing was done with a classic Integrated Development Environment (IDE), such as Visual Studio, on a PC with Windows. There was simply a folder mapped to the Docker container. To perform actions in the Docker container, it had to be accessed from the Windows terminal. There was also a possibility of showing all the figures with an Xlaunch app. The Linux with a Docker container was run on a computer with Nvidia Gtx1060 6Gb, which was already a rather old, low to mid-end performance GPU but sufficient for the purposes.

Dataset

First, a dataset that would be used for training had to be created. As mentioned before, there was a damper with two defects and a metal part with one defect to be inspected. To make a dataset, there was a universal procedure developed.

It starts with placing the defected part to the clamping mechanism, turning the lights on, and adjusting the position of the camera. First of all, it needs to be decided what the best angle for the camera is, ensuring the defect visibility during rotation, with the biggest contrast possible. This requires some experience on the user side. After it is established, the camera is adjusted using the positioning system described in the subsection 4.1.2.

The next step is to open the Basler pylon desktop application and further adjust the position of the part so that it covers most of the field of view and is centred. After that, the camera positioning mechanism is fixed, and the lens adjustment starts. It is imperative to set the focus so that the object is as sharp as possible in the image.

The next thing to do is adjust the aperture of the lens to achieve optimal brightness and a high contrast of the image. This needs to be done hand in hand with exposure settings. The goal is to find the balance between the depth of view and the sharpness of a moving object. After that, lights should be adjusted to highlight the defects. The adjustment is carried out using four laboratory power sources, each for an individual wall. The final setup is always the result of a few iterations, because changing one setting impacts other optical parameters, so the user has to return to adjusting the previous setting.

All the inspected parts shared the same exposure time of 3 ms because they all rotated at the same speed. It could have been set higher without causing blur, but there was no need for a large depth of field. Both the damper and the metal part shared the same lens with a 8 mm focal length and its setting, as well as the camera position, which can be seen in the picture 4.20. This could be so as they were both rotational and the size difference was insignificant. The light conditions were nevertheless very different.

The integrated light system came handy when adjusting the light conditions in order to make an ideal environment. For the damper inspection the left wall shined a lot (11.5 V), both the back and front wall were set to shine at a medium level (9 V) and the right wall was completely off. The sides are defined from the perspective of a user looking inside the box through the door. This setting created a great contrast between the background and the object, as shown in the picture 4.22. Also, the camera was positioned so that the defect on the bottom skirts was visible against the bright background, which proved very beneficial, as can be seen in the picture 4.21. It might appear that the images have a green tone, but this is due to the white balance being turned off. This had nonetheless no effect from the perspective of CNN.

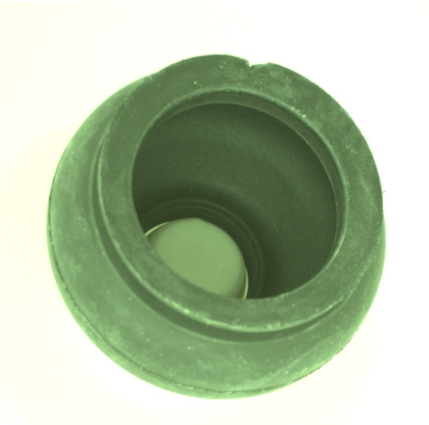


Figure 4.21: First defect dataset sample

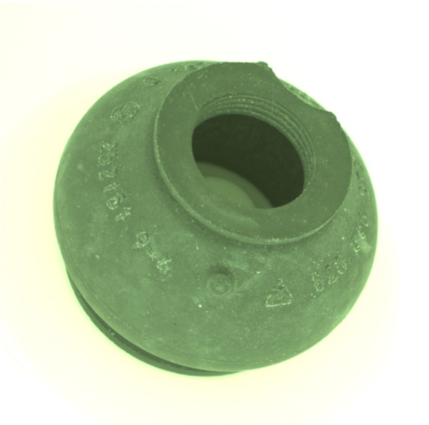


Figure 4.22: Second defect dataset sample

The metal part had each of the light set to a low-level shine creating a dim uniform light. This made sense because it was mostly colour that was inspected in this application. The sample picture of the defect parts dataset can be seen in the picture 4.23 and the sample picture of the correct parts dataset can be seen in the picture 4.24.

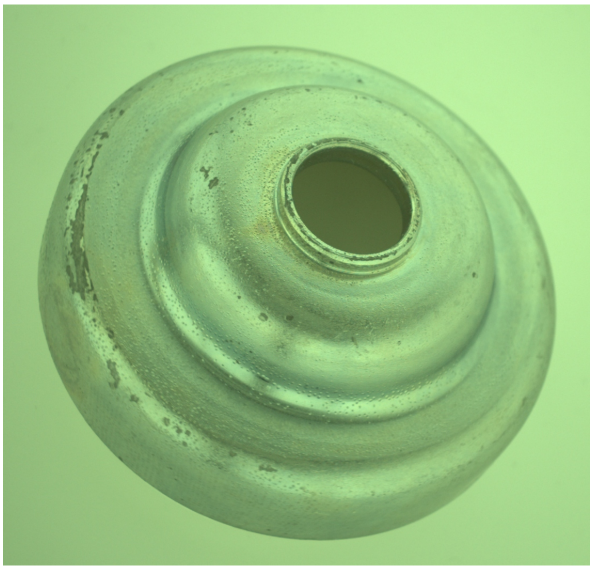


Figure 4.23: Metal part dataset defect sample

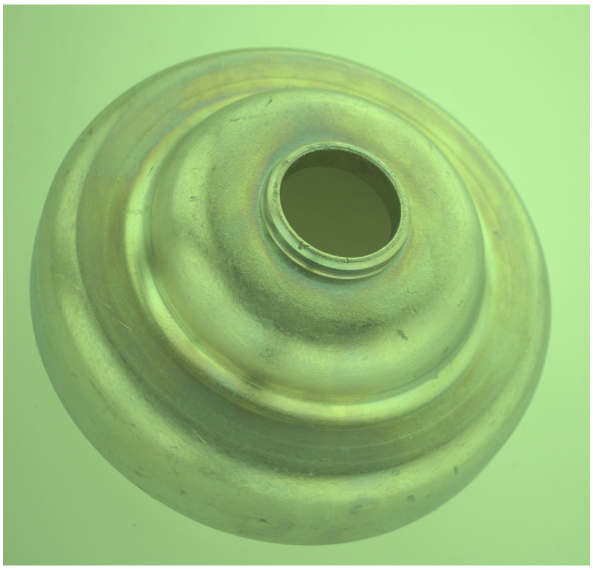


Figure 4.24: Metal part dataset correct sample

Although the coffee cups were not used for training, some sample images were taken to demonstrate the usefulness of the integrated lighting and to test the alternative side attachment of the camera. Also, a lens with a focal length of 12 mm was used in this case with a different setting. My personal opinion is that the integrated lighting worked really well in this case, as it allowed creating conditions that made the edges visible. Furthermore, there were no major reflections, which is generally difficult to achieve when working with glass objects. The sample can be seen in the picture 4.25.

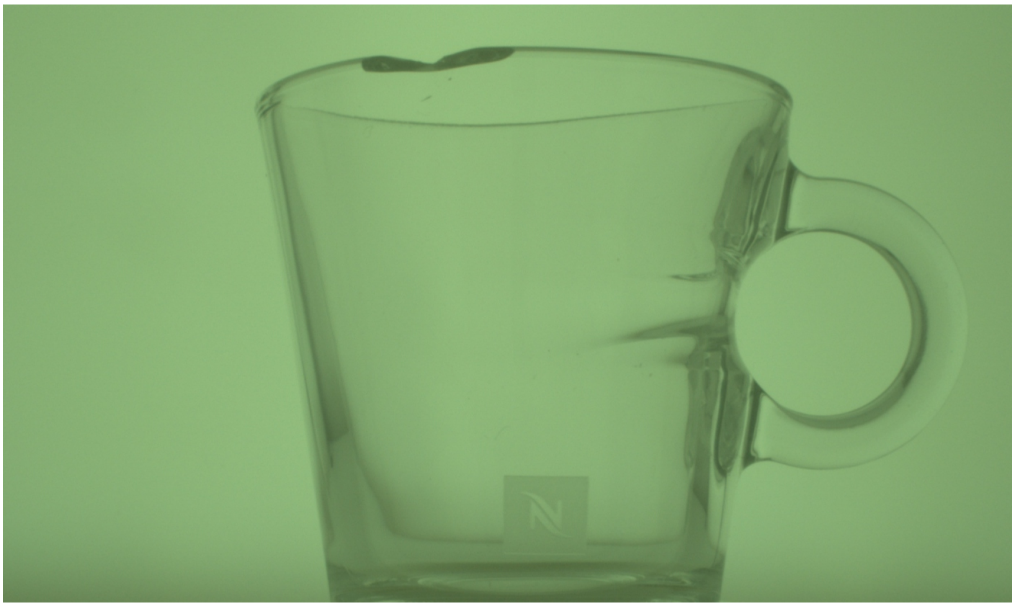


Figure 4.25: Sample image of a glass coffee cup

The next step of dataset creation procedure, after achieving the ideal optical system setup, is to run the Python script for creating a dataset. It contains the part which handles the camera control and was described in subsection 4.2.2. In addition to that, whenever an image is captured, it stores the images in a labelled folder using the OpenCV library.

This script is part of the attachments and is called "dataset_create.py".

To clarify, it is necessary to manually place sample parts to the optical cell and rotate them using the PLC programme. Twelve images are captured and stored per single rotation. After going through all the sample parts, the dataset is created. These images undergo a slight Joint Photographic Experts Group (JPEG) compression, which reduces their size, but other than that they are unedited. Sometimes it might be necessary to crop them.

There was an additional Python script developed for this purpose. It simply goes through the images, crops them and then stores them in an additional labelled folder. This script is a part of the attachments as well and is called "dataset_edit.py".

As mentioned earlier, there were 50 correct sample parts and 50 sample parts of each defect of dampers. 10 samples from each category were moved aside to be used for the testing, which means that only 40 sample parts from each category were used for the dataset creation. In the end, it contained approximately 480 pictures of each category of dampers. The dataset of metal parts contained approximately 120 pictures of each category.

Architecture

The next crucial step was to design an applicable tool for defect detection. A classification convolutional neural network was chosen for this purpose because it is a great tool for this. It is highly uncommon to design a CNN or even a NN from scratch on one's own. Prevalently, a predefined architecture along with pre-trained weights is employed and the classification part of the neural network is simply adjusted. After that, the architecture is complete and it is only necessary to further train the network with the created dataset.

In our case, it was decided to use the Keras API to download the InceptionV3 network, which was described in subsection 3.3.3. This architecture was chosen for its high accuracy to the prediction time ratio, as can be seen in the picture 3.56. Furthermore, it was decided to use the version with the weights, which were pre-trained using the ImageNet dataset. This network has an RGB image with the size of 299x299 px as a default input. Therefore, the input layer had to be changed in order to use an image of a different size as input. This was an easy task as the input shape was an optional parameter of the Keras function for InceptionV3.

This network has 1000 classes by default. It was decided that two classes representing the correct and faulty parts would be sufficient for this application. Thus, the top of the network, meaning the classification part that comes after the convolutional part, was not included. This was achieved by setting a parameter of the Keras function for InceptionV3.

Instead, a GlobalAveragePooling2D layer was added, which served as a flatten layer. After that, a fully connected layer with the ReLu activation function was placed. Lastly, there was a classification layer with the softmax activation function and two outputs added. This architecture was inspired by a tutorial on the Tensorflow website [52] and is quite typical.

Training

In order to make a CNN that would be accurate in the prediction of defects, it was essential to train it. Let us suppose that we already had a dataset and an architecture designed following the steps described in the previous subsections.

The data fed to a CNN must have a certain format. It is typically a tensor, which is a multidimensional object containing matrices of images along with labels. But with Tensorflow, there are many methods for how to achieve it. Each method produces a slightly different type of object that could potentially be used for either training or prediction. On top of that, the newer versions of Tensorflow enable the use of additional methods.

There was a Python script made for training purposes, which is part of the attachment and is called "cnn_train.py". It also contains the architecture. The first thing this script does is that it loads and preprocesses the dataset. As was said, it can be done multiple ways, but it was decided to use a Keras function called "image_dataset_from_directory", which is easy to use and does a lot of things automatically.

This function creates a dataset object that has images loaded from an assigned directory and labels them according to the name of the folders. This means that the path should lead to a directory with two folders, where one contains the images of the correct parts and the other the images of the faulty parts.

Furthermore, this function offers many optional features. In this script, the 20% validation split is used, which means that 20% of the dataset is used for validation purposes, shuffling is used, which randomly mixes the images to produce a diverse dataset. In addition, the images are resized to reduce the subsequent computing power requirement. This is not optimal, but it has to be done. There must be a compromise between the resolution and the computing power requirement.

The last thing that is done is dividing the dataset into batches, where each contains a defined number of images, 10 in our case. These batches serve as an input to CNN, where the whole batch is handled in one iteration.

Once this function is applied, the object is created and could already serve as input for training. But to further improve the dataset, normalisation is performed. It is done by dividing each element of image matrices by 255. This leads to a situation where all the matrices have their elements in the range from 0 to 1. After this, we have a preprocessed dataset. Data augmentation, which would further extend the dataset, could also be done. However, it usually works by distorting the image, which would in our case produce unrealistic results. That is why it was not used.

Before the training process starts, it is necessary to select an optimiser that defines the method of backpropagation. In this application, Adaptive Moment Estimation (Adam) optimiser was selected. It is one of the more advanced methods and generally offers very good results. Next, the loss function needs to be selected. In this case, the sparse categorical cross-entropy loss function is used because it is often used for such application.

Once all these settings are set, we are able to start the training process. It is managed by a Keras method "fit", which has a dataset including the validation split as an input, as well as the batch size and number of epochs. The number of epochs defines how many times the whole dataset is fed into the network during training. In addition, shuffling is used as an optional parameter. This is redundant, as it was already done when the dataset object was created, but it does not cause any harm.

After the training process finishes, we use the Keras function "save"to save the trained

CNN as a file with the ".h5"suffix, which is a format typical for CNN.

The progress of the training can be displayed utilising various metrics. In our case, accuracy, validation accuracy, loss function, and validation loss function were monitored after every epoch. The graphs showing the dependency between these metrics and epochs during training of the first type of damper defect can be seen in the picture 4.26. Accuracy defines the prediction precision on the data used for training, while validation accuracy shows the prediction precision on the validation data. Loss and validation loss define the difference between the prediction and the real value when using training and validation data.

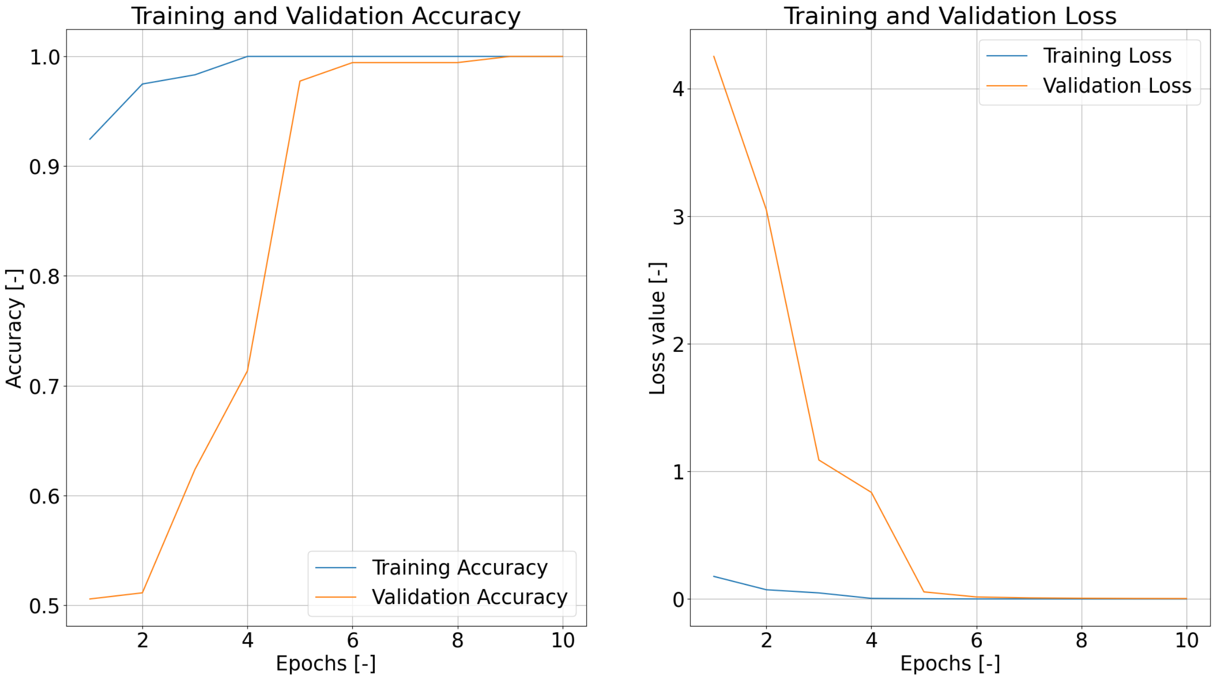


Figure 4.26: Dependency of training metrics on epochs for first type of damper defect

The training accuracy is naturally very high even after just one epoch. This is because the defects are not diverse at all, look more or less the same, and the dataset is relatively large. This means that the CNN receives a similar input many times per epoch.

A more interesting metric is the validation accuracy since it gives information on the real prediction precision. We can see that the CNN was able to correctly predict almost 100% of the validation data after 6 epochs. Once the validation accuracy was precise enough, the training could be stopped. To continue in training would mean risking overfitting. That is why the final CNN used for testing was run through 7 epochs during training.

There was a completely new CNN trained for each defect, using the same methodology. In total, there were 3 CNNs created. One for each damper defect and one for the metal part.

4.3 Design of the robotic workplace

This section aims to describe the creation of a workplace. It gives a detailed look at the 2-DOF robot, which was designed as a part of the workplace and serves for moving the carriage. After that, it discusses the concept of the workplace, as well as the layout and used components. Lastly, it deals with the functioning of the whole workplace and with the communication between the PC, the PLC and the Aubo i-5 cobot. The whole workplace is supposed to serve the marketing and testing purposes of Kinalisoft s.r.o. and is not to be sold to a customer. It was thereby not designed to comply with the norms.

4.3.1 Two degrees of freedom robot

The 2-DOF robot was mentioned several times, but never discussed in detail. That is why this subsection is focused on its design and functioning, from both hardware and software perspective. The robot was inspired by the previous Test-it-off workplace but it was completely redesigned including the use of different electric motors.

Design

It was decided that the axes used to build the robot would be ordered from the company Shenzhen W-robot Industry Co., Ltd., which offers a wide range of axes. The products of this manufacturer had been used before without any issues. These axes use an electric servomotor to rotate the motion screw, which moves the carrier, sliding on the guidance.

To choose the right model, calculations had to be done. As was already mentioned, the maximum weight of an object was established to be 5 kg. The weight of the carriage was approximately 5 kg as well. For that reason, the carrying capacity of at least 15 kg was opted for in order to have a slightly excessive setup. After establishing this parameter, it was clear that the series with a 12 mm diameter of the motion screw had to be used. Another parameter to be set was the pitch of the screw motion, which dictates the distance of linear movement per rotation. Using the datasheet, the 16 mm pitch was selected for the horizontal axis and the 10 mm pitch for the vertical one. A different pitch for each axis was used because the vertical axis moves against the gravity force and therefore a bigger force is applied on the screw. Next, the length of the axes had to be chosen. The required stroke of the horizontal axis was calculated based on the inner dimension of the box in the direction of carriage movement, which was 310 mm. It was not necessary to move the whole carriage from the box because the clamping mechanism was positioned in the middle of it. Hence it would be sufficient to move the carriage only to a position where the clamping mechanism is accessible by the collaborative robot. It was nonetheless decided to select the length of 420 mm, since there was enough room inside the work table and the price difference was insignificant. This way there was enough stroke to move the whole carriage out of the box and keep a reserve.

After that, the stroke of the vertical axis had to be computed. This choice depended on the size of the box gap for the carriage to go under the door, which was 300 mm. However, the carriage was 100 mm thick. Therefore, 200 mm would be sufficient stroke to close the gap, but the carriage was required to go further up to get to the area of lighting and to leave room for the distance between the object and camera adjustment. That is why a 300 mm stroke axis was chosen.

Both of these axes are driven by 400 W Panasonic servomotors. These electric motors are excessive, but it is better to have more powerful motors, as they allow for higher accelerations. Furthermore, each axis has two optical sensors for the homing and position limit purposes. The selected axes can be seen in the picture 4.27 below.

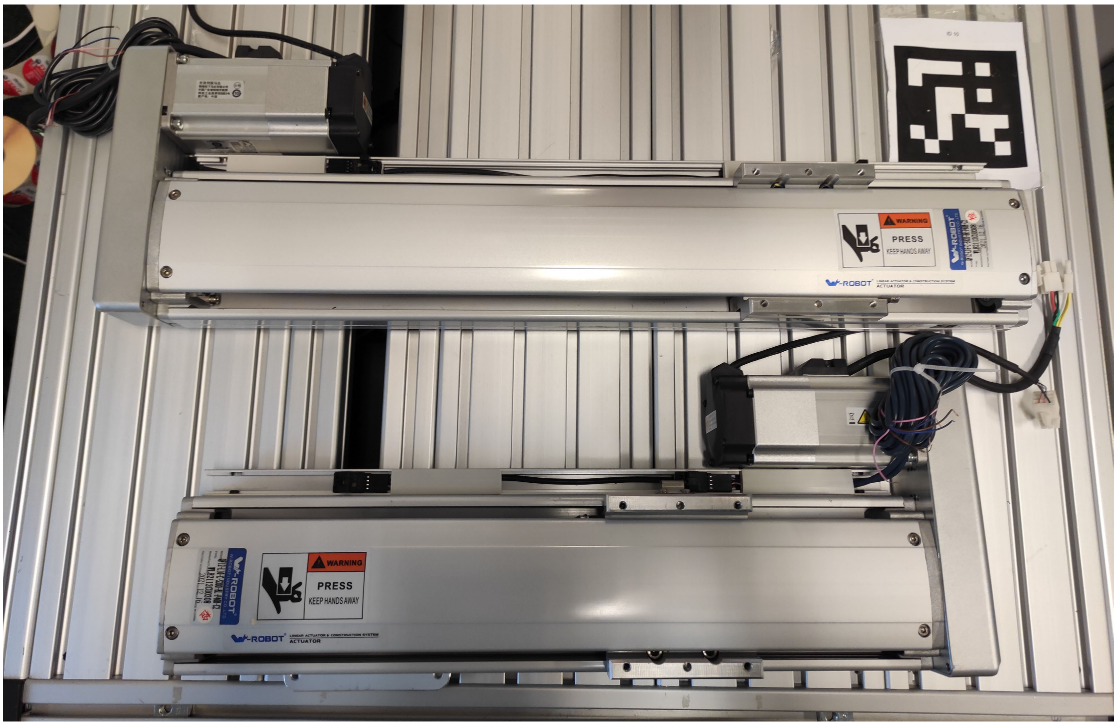


Figure 4.27: The selected axes for 2-DOF robot

Having secured the right axes, the next step was to decide how to form them into the 2-DOF robot enabling the attachment of either the carriage or the gripper. For that purpose, three aluminium machined plates were designed. The first served to attach the horizontal axis to the working table. This was achieved by angle brackets connecting the plate with the bottom surface of the work table desk. The second one simply served for attachment of the vertical axis to the slider of the horizontal axis.

The third aluminium plate was attached to the slider of the vertical axis and carried either the gripper or the carriage. Its length was calculated, so that when the slider was at the vertical axis bottom limit, the bottom of the carriage touched the tabletop. Next, a canal for wires was designed to create space for leading the wires from either the carriage or the gripper. This canal was covered by a steel sheet, which was designed for this purpose. The 3D model of the 2-DOF robot can be seen in the picture 4.3 and the picture 4.28 of the real robot can be seen below.

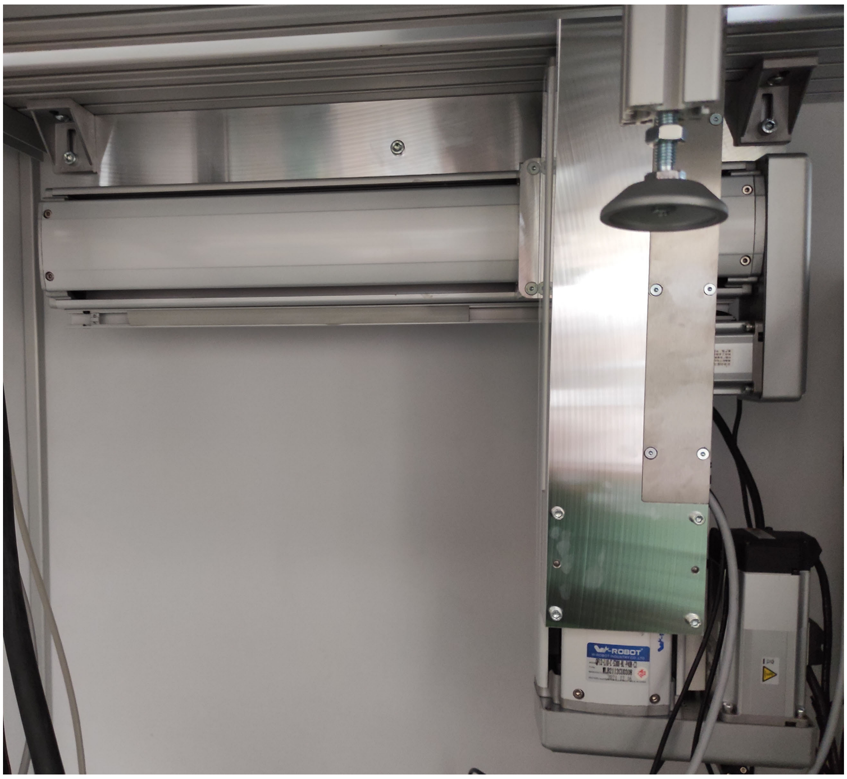


Figure 4.28: 2-DOF robot

Control

As mentioned earlier, the axes are driven using Panasonic servomotors, the Minas A6 family motors to be precise. These electric motors are controlled by very complicated drivers that offer an immense number of options. These drivers along with many cables were delivered in the package with the axes. It was a straightforward task to connect most of the cables, including those for brake release, motor powering, and encoders.

There was also a 50-pin connector that serves for communication purposes between the PLC and the driver. There was no cable delivered, just a connector. The datasheet clarified that there was a bus with 50 inputs and some of those had a very specific purpose, useful only in specific situations.

In addition, the driver allows for many different ways of controlling communication. The easiest turned out to be the position control using the command pulses protocol. This method employs two wires to transmit pulses. The movement of the axis is based on the combination of pulses on both wires. There are multiple options of combination logic. The most straightforward one was selected, which uses one wire for the position purpose, and each pulse means that the motor rotates within the defined angle. The second wire is used for direction purpose and works with the two-value logic. Hence, when the logical value on the wire is true, the motor rotates in a certain direction, and when the value is false, it rotates in the opposite direction.

After establishing the communication protocol, it was clear which pins of the 50-pin connector would be used. It was necessary to use 9 pins to control the axis without the utilisation of any additional features. However, some of these features, such as the alarm and alarm reset, could be handy. Because of this, a 16-wire cable was opted for. After choosing the features to be used, the cable was soldered to the connector, as shown in the picture 4.29.

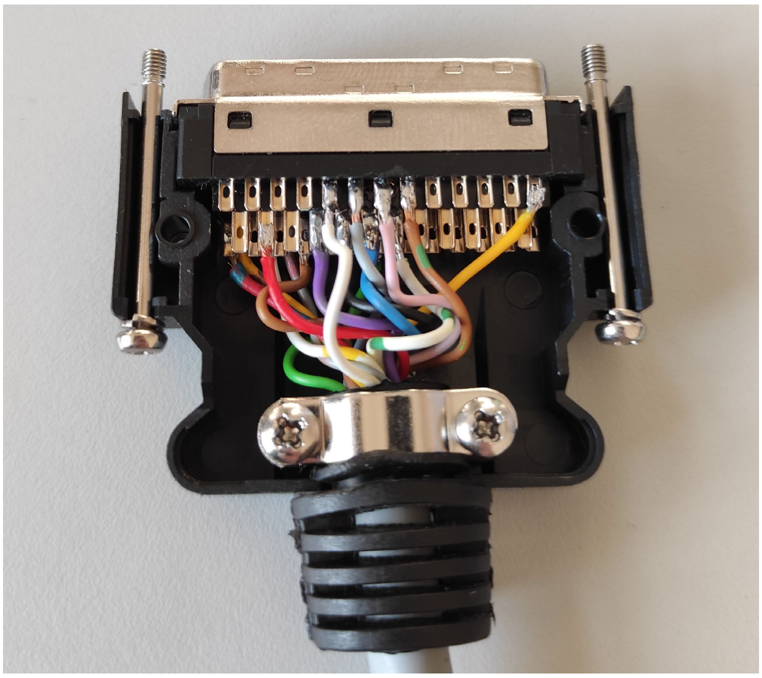


Figure 4.29: 50-pin connector

Once the electrical part was complete, it was necessary to program the PLC to control the 2-DOF robot. There was nonetheless still one thing to be done. The drivers had to be set up using the Panasonic's Panaterm application. To do this, each driver had to be connected to the PC via USB. After that, it was easy to get connected using the app, which offers a lot of functions, such as fine-tuning, monitoring, controlling, etc. In this case most of the options were left default. Only a few of them were changed, such as the choice of communication, which was set to the previously mentioned command pulses, the number of pulses per rotation, which was set to 3600 to allow the position regulation with 0.1° precision, etc.

After setting up the drivers, it was finally possible to make a PLC program and test the 2-DOF robot. Siemens S7-1200 1214C DC/DC/DC PLC was decided to be used along with an additional Siemens DC module because it met the requirements and was available. It was programmed using TIA Portal V16, which is a standard Siemens PLC programming tool.

Thanks to the Siemens motion library, it was very easy to add the axes and configure them. This could be done in a few steps, by selecting the type of communication, hardware parameters, and by assigning the Input/Output (I/O). After that the object representing the axis was added and the axis could be controlled manually, which is practical for the testing.

It was necessary not only to add the axes, but to use them in a program. Therefore, a function controlling the axis was created for each of them. This function was written using the Ladder logic (LAD) language, which uses function blocks connected sequentially. When calling this function, a series of commands is executed one after another.

To briefly explain the logic of the function, there is a list of execution variables, such as one for homing, relative or absolute move, etc., and value variables that define the position and speed. All these variables are the input of the function and control which commands are executed. Importantly, to allow execution of move commands, the axes have to be homed.

The homing procedure starts by checking whether the signal from the homing sensor is true or not. If it is true, it means that the axis is in close proximity to the homing position. For that reason, a relative move is executed with the aim of moving away from the sensor. Once the sensor signal is false, the axis stops and the homing function block is executed, which starts the homing. After it is finished, the corresponding variable is set. Then it is possible to control the axis by setting a value of position and speed and executing either a relative or absolute move. There is also a variable for executing the stop of the axis. These functions for controlling the axes are used in the main program afterwards.

4.3.2 Two modes concept

One of the thesis objectives was to integrate the optical cell into a robotic workplace. Because of the circumstances, it was decided that a whole new workplace would be built. This workspace was nevertheless not supposed to serve only the purpose of an optical cell but also for a Test-it-off product, which is a smart robotic workspace for PCB testing, including an advanced software platform.

Because of this, a two-mode concept was born. The workplace was to be developed using an already built work table. Its desk was adjusted to make a gap for the 2-DOF robot. An Aubo i-5 cobot was chosen to be a part of the workplace and to serve for pick and place purposes. The initial state can be seen in the picture 4.30.

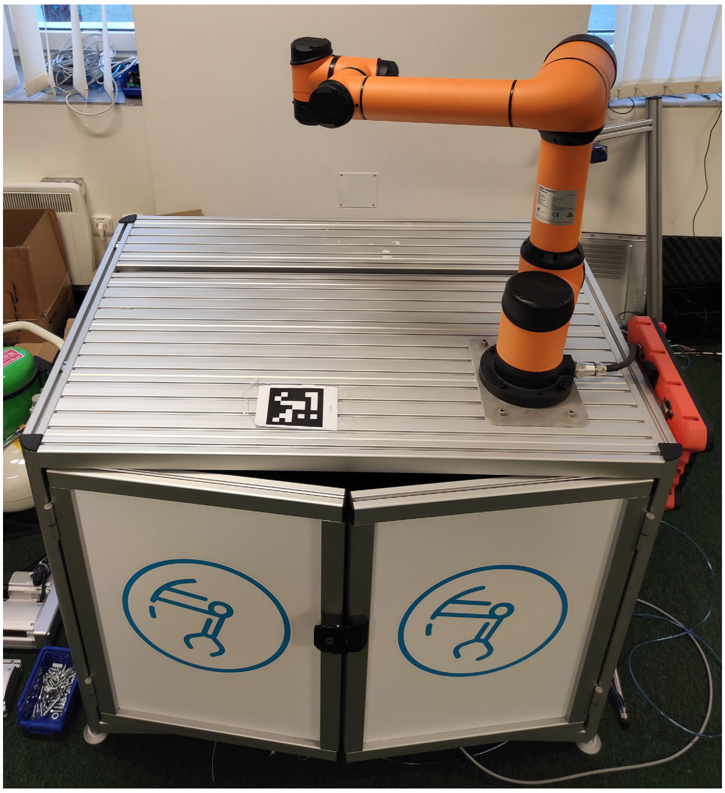


Figure 4.30: Initial state of the workplace

The first thing that needed to be done was to come up with a concept. There had already been one Test-it-off workstation built, which can be seen in the picture 1.1, and served as an inspiration. This workplace contained a PCB tester, a 2-DOF robot with a gripper called a PCB placer as it was used for the precise placement of the PCBs to the tester. In addition, there was a 3D camera for bin-picking. Three boxes served as containers, one for the input of material, one for the defected output and the last one for the correct output. They were fixed by frames. The newly built workplace had to contain all these parts. Yet, their design had to be modified.

First of all, the distribution of components on the new workplace had to be established. As was already mentioned, the workplace should serve primarily for trade fair purposes. With this in mind, a side without a door was selected to be the front side. Ideally, the potential customer could see the most interesting actions taking place closer to the front side. For that reason, the optical unit as well as the tester had to be put close to the front side. This way, the 2-DOF robot had to be placed there as well.

Inside the table, it was established that there would be the axes of the 2-DOF robot, the Aubo i-5 control box, an electrical distributor, and a PC. Thus, it was advantageous to have the 2-DOF placed on the opposite side of the doors because there was enough room for the control box and PC to be positioned near the door to be accessible.

The idea behind the two-mode concept was that there would be either the tester or the optical box and that the 2-DOF robot would carry either the gripper or the carriage. There was a frame developed from bent steel sheets for positioning of the box. The tester is attached in its own positioning mechanism that has outer dimensions equal to those of the optical box. The tester can thus be switched with the optical box without a need to readjust it every time. There were also frames for fixing the boxes built from steel sheets. All these parts were newly designed and are different from the ones used in the previous Test-it-off workplace. The mode for Test-it-off can be seen in the picture 4.31 and the mode for optical cell can be seen in the picture 4.32.

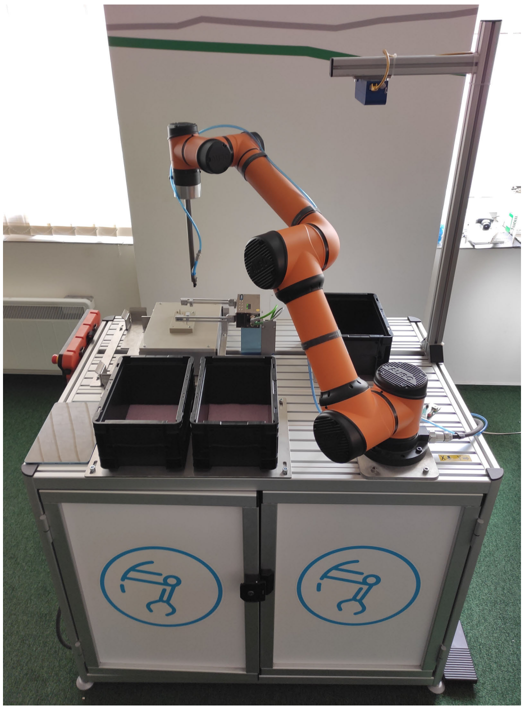


Figure 4.31: Test-it-off mode

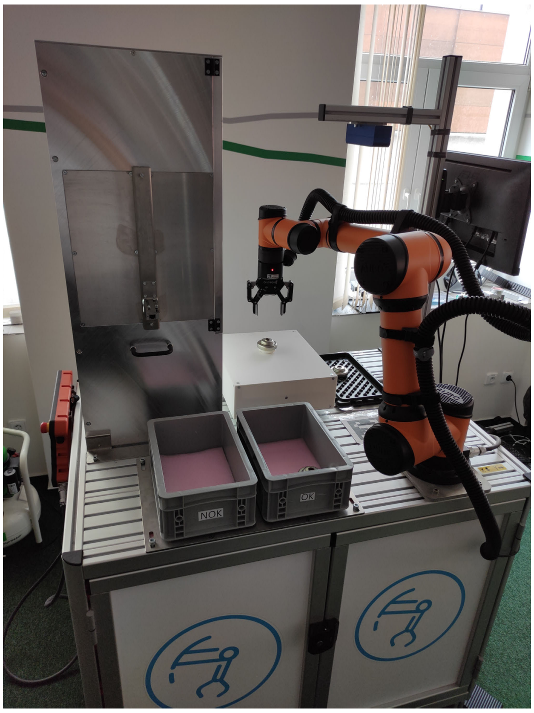


Figure 4.32: Optical cell mode

Also, the end effectors are different for each mode. There is a vacuum end effector used for manipulation of PCBs and a Robotiq gripper used for manipulation of dampers and metal parts.

The last crucial part that needed to be added to the workplace was an electrical distributor. Its main purpose was to accommodate the drivers and the PLC. Furthermore, the I/O of the PLC, +24 VDC and the ground was distributed to the terminal plate, because it is much more convenient to use it this way. The electrical project was created by an external office. However, the concept design and the construction were carried out as part of this thesis. The picture 4.33 of the electric distributor inside can be seen below.

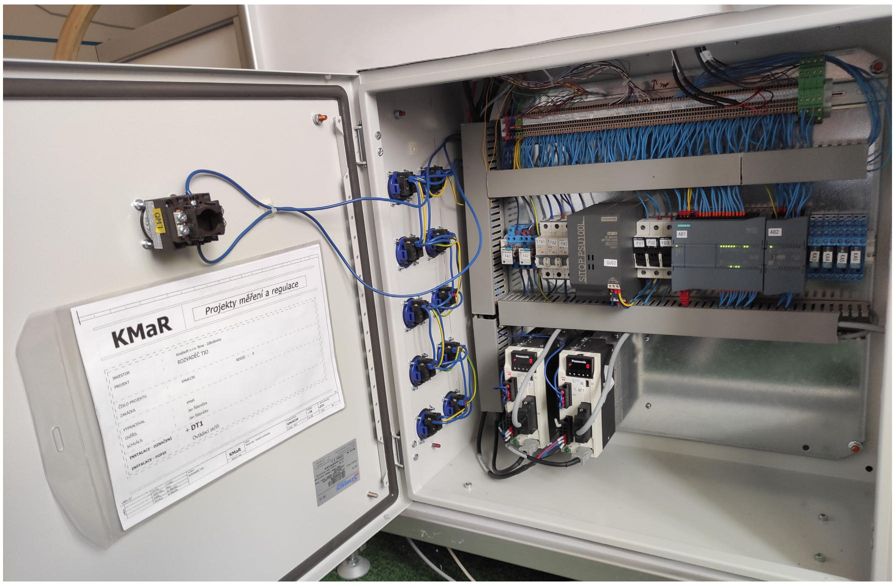


Figure 4.33: The electrical distributor

Most of the I/O of the PLC were used by the wires for communication with Panasonic drivers. The connection of the sensors was quite straightforward and needs not to be described. The only interesting thing is that the optical sensors of the axes used for homing are NPN, yet the PLC is designed for PNP inputs. To solve this issue, an NPN to PNP converters had to be added. Another interesting design feature are the relays that can be disconnected using the stop button, which leads to shutting down the gripper and the axes.

There was also a 2-phase digital stepper driver for the rotational system installed. The precise model was Leadshine DM542EU. It is controlled from the PLC the same way that the Panasonic drivers are controlled but it is much simpler and does not have any additional features.

4.3.3 Functioning of the workplace

The workplace was described and a detailed look at the main components was given. Yet to make the workplace work as a whole, logic had to be developed and communication between devices established. This subsection describes how the workplace functions when in the optical cell mode. The Test-it-off mode works in a completely different way, not relevant to this project.

There clearly had to be communication between the PC, the Aubo i-5 cobot, and the PLC. The easiest way to do this turned out to be to create Modbus TCP communication. The idea was that the PLC would act as a server and communicate with both the PC and the cobot. Creating a server using a PLC is really easy because it is done with a single function block. To make a register, which would be used for writing down the values for communication, an array of words is created. There had to be two servers made because only one device can be connected to one port. Because of that, one server is used for communication with the PC and the other for communication with the cobot. When these devices are connected, they can edit the values in the register the same way as PLC.

The Aubo i-5 has a Modbus plugin, and to establish communication, it is only necessary to fill out the Internet Protocol (IP) address of the PLC and the port number. After the cobot connects to the PLC, it is possible to define variables corresponding to the particular addresses in the register. These variables can be used for decision making in the cobot program and for providing the PLC with the information about the state of the cobot.

A Python script was developed for testing purposes. There was a Pymodbus library used to establish communication between the PC and the PLC, which was very easy to use. This script is a part of the attachments and is called "cnn_test.py".

The first part of the script serves the initialisation purpose. It starts with connecting to the PLC and establishing Modbus communication. Then the trained CNN is loaded, the camera is connected and its settings are adjusted as was already discussed. Once this is done, the script ends up in a never-ending loop. Whenever the images are captured, a batch is created. Then it is fed to the CNN, which predicts the probability of each image showing either a correct or a faulty part. There is a decision-making algorithm that determines that a part is faulty whenever any of the values representing the probability of part being faulty is above 0.5. This means that if only one image is predicted to be faulty with the probability above 50%, the part is tagged as faulty. This final verdict is written as a defined value for a certain address on Modbus server.

In this manner the PLC gathers information about the result of the defect detection algorithm as well as controls and monitors the cobot. There naturally needs to be a main program controlling the whole logic of the workplace. A PLC program was created for this purpose. It contains the functions for axes control, Modbus server establishment and a state machine for the decision logic. The state machine acts like a brain of the entire workplace. The functioning of it can be seen in the mind map shown in the picture 4.34 below.

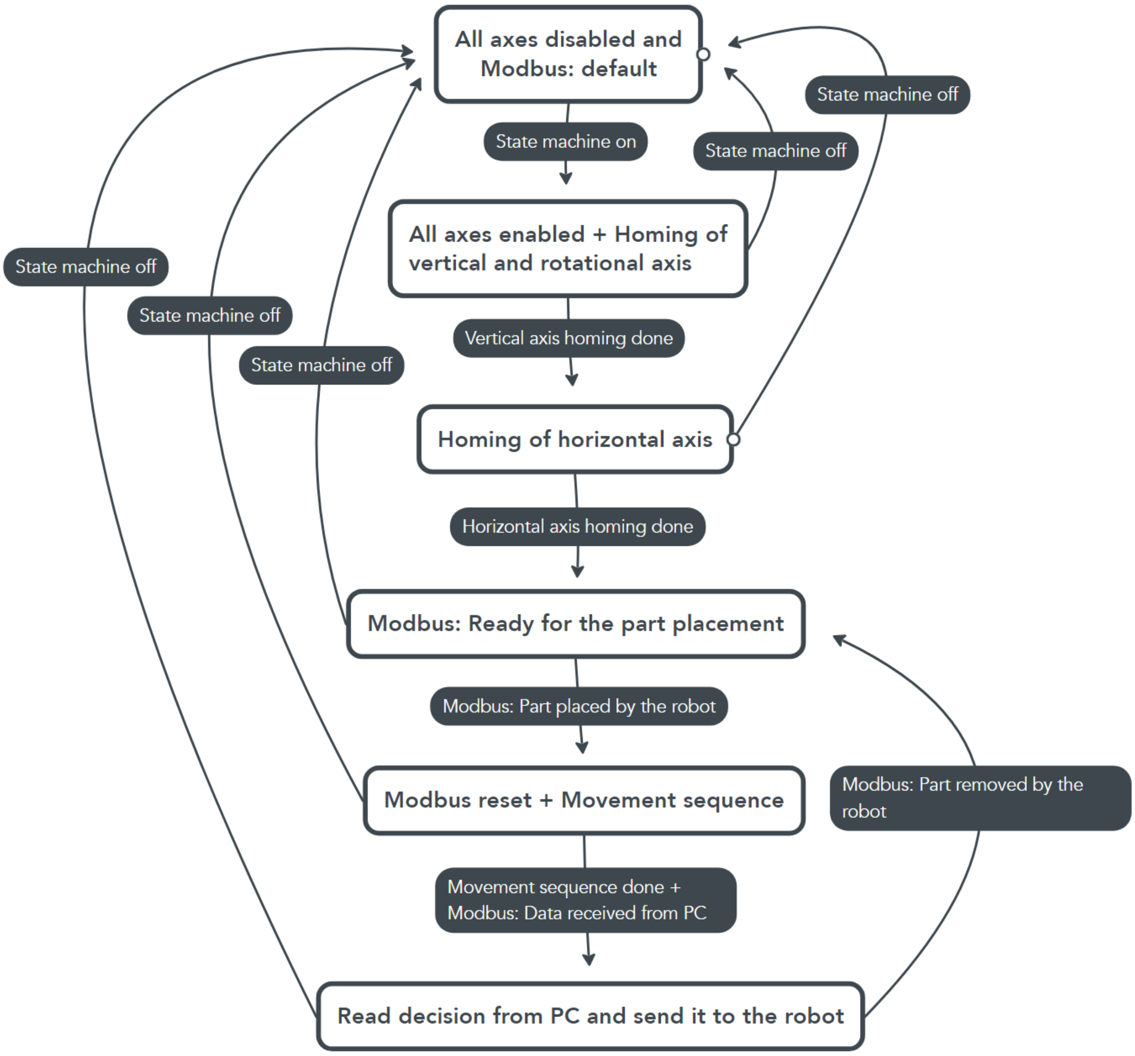

All axes disabled and Modbus: default
State machine on
State machine off
State machine off
All axes enabled + Homing of vertical and rotational axis
State machine off
State machine off
State machine off
Vertical axis homing done
Homing of horizontal axis
Horizontal axis homing done
Modbus: Ready for the part placement
Modbus: Part placed by the robot
Modbus reset + Movement sequence
Modbus: Part removed by the robot
Movement sequence done + Modbus: Data received from PC
Read decision from PC and send it to the robot

Figure 4.34: Mind map of the state machine

4.4 Testing

The design of the optical cell and the functioning of the workplace was described as well as the software for defects detection. Tests were nevertheless necessary to determine how it works together and how precisely it detects defects. The best way to test it seemed to let it run for a while and monitor how many mistakes it makes.

At first, the accuracy of the damper quality inspection was to be tested. As already mentioned, 10 pieces of correct parts and 10 pieces of parts with each type of defect had not been used for training. These 30 parts were to run through the optical cell twice and the number of mistakes was to be written down. It also seemed interesting to make a comparison between the accuracy of optical inspection of the parts used for training and those that were not. That is why another set of 30 parts with the same defects distribution was selected from the ones used for training and was tested the same way. Two tables can be seen below, where table 4.1 represents the statistics when testing the parts that were not used for the training and the table 4.2 when using the ones that were part of the training dataset.

Type of the part	**Amount of parts**	**Correctly detected parts**	**Incorrectly detected parts**	**Accuracy**
Correct part	20	20	0	100%
First type faulty part	20	20	0	100%
Second type faulty part	20	20	0	100%

Table 4.1: Statistics of the optical inspection of dampers of parts not used for training

Type of the part	**Amount of parts**	**Correctly detected parts**	**Incorrectly detected parts**	**Accuracy**
Correct part	20	20	0	100%
First type faulty part	20	20	0	100%
Second type faulty part	20	20	0	100%

Table 4.2: Statistics of the optical inspection of damper parts used for training

We can see that the results were the same. Both tests achieved 100% accuracy. This was expected because, as was already said, there was little diversity in the defects, meaning that the parts used for training looked very similar to the ones not used for training. Still, the result is very good and shows that the concept works well.

Once the damper testing was finished, it was decided to test the surface treatment of the metal parts as well. The methodology was almost the same. The only difference was that all parts had been used for training because there was a lower number of sample parts. It was still decided to let all 20 parts run through the machine twice. The test statistics are shown in the table 4.3 below.

Type of the part	Amount of parts	Correctly detected parts	Incorrectly detected parts	Accuracy
Correct part	20	17	3	85%
Faulty part	20	20	0	100%

Table 4.3: Statistics of the optical inspection of dampers of parts not used for training

The results show that this time the optical inspection was not always accurate. One of the parts, which was supposed to be correct, was classified as faulty in both runs. On top of that, one more correct part was classified as faulty during the first run but not during the second one. Both these parts had nevertheless their surface colour almost similar to the faulty one, which indicates that the surface treatment was not the best. These parts may thus be considered marginal. In addition, it was discovered that the probability of the part being defective was slightly above 50% for just a single image. However, this is sufficient for the algorithm to tag it as faulty. The difference between the marginal correct part and the defective part can be seen in the picture 4.35. Clearly, there is only small difference and that is why it is difficult for the CNN to decide whether it is correct or not.

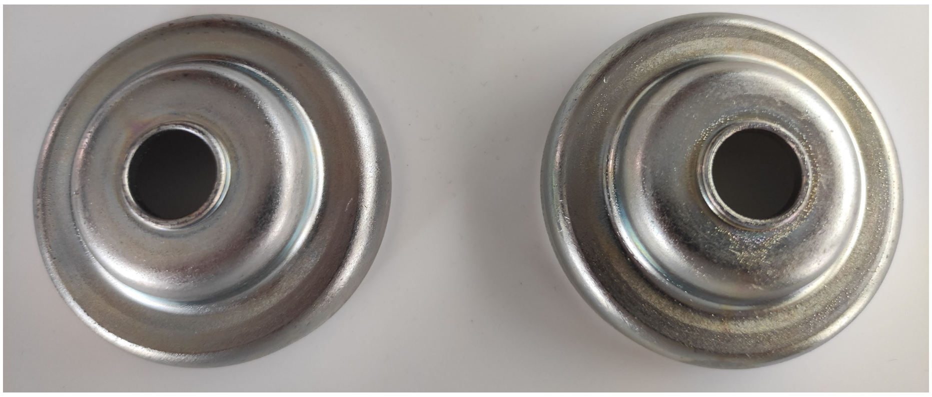


Figure 4.35: The difference between two marginal objects each belonging to different category

Except for that, the optical inspection provided a consistent and accurate prediction for parts that clearly belonged to one category. The average accuracy of detecting the correct and the faulty parts was 92.5%. That is why it could be stated the optical inspection of the surface treatment of metal parts worked decently.

5 Conclusion

A universal cell for optical part inspection was developed and built as part of the project described in this thesis. Initially, a concept of the carriage, which goes in and out of the box, was invented. The box was made from aluminium plates and contains integrated lighting made of LED strips that can be regulated. It is located behind diffusive plexiglass boards. The positioning mechanism was developed, which allows the user to adjust the camera according to the application.

The carriage includes a rotational mechanism that is used to rotate the part during the image capture. In addition, a camera triggering mechanism consisting of a coding disc and inductive sensors is a part of the rotational system. A clamping mechanism that contains a flange and mandrels was developed. There is a specific mandrel for each part and it is very fast and easy to switch between them.

A camera and a lens were selected. To control the camera, a Python script was written, which works hand in hand with the rotational system and the PLC. The software for defect detection was developed, which uses a pre-trained CNN. A system of Python scripts was made in order to quickly and efficiently create a dataset, train the CNN, and test it.

As part of this project, a whole new workplace was designed and built. It works in two modes. The first one is for Test-it-off PCB testing and the second is for the optical cell. The workplace contains the 2-DOF manipulator, which was freshly designed and programmed. It serves to carry either the carriage or the gripper. Furthermore, the Aubo i-5 robot is part of the workplace. In order to make the workplace function, a state machine logic was developed that uses the Modbus protocol to establish the communication between the PLC, the PC and the Aubo i-5 robot.

Two unlike parts were chosen to test the functioning of the whole product. The first was a damper, which had two types of defects, and the second was a metal part with one type of defect. There were CNNs trained using the system of Python scripts and the developed methodology. The testing consisted of parts being run through the workplace and of monitoring the mistakes. The final results confirmed the correct functioning of the product, since the dampers inspection had 100% accuracy and the metal part inspection 92.5% accuracy, which was caused by an unclear line between the correct and faulty part.

The system still needs to be integrated into the Test-it-off concept and there is room for improvement. Nonetheless, a well-functioning complex system was built that meets all the objectives of the thesis.

Bibliography

[1] Automatizujte kontrolu výrobků pomocí robotického systému Test-it-off. In: *Kinali* [online]. [cit. 2022-05-15]. Available at: https://www.kinali.cz/cs/clanky/automatizujte-kontrolu-vyrobku-pomoci-robotickeho-systemu-test-it-off/

[2] Machine vision. In: *Wikipedia* [online]. San Francisco (CA): Wikimedia Foundation, 2022 [cit. 2022-04-10]. Available at: https://en.wikipedia.org/wiki/Machine_vision

[3] Rule-Based Vs Machine Learning Approach. In: *Qualitas Technologies* [online]. September 10, 2020 [cit. 2022-04-15]. Available at: https://qualitastech.com/blog/image-processing/differences-between-machine-learning-and-rule-based-systems/

[4] CHOOSING BETWEEN MACHINE VISION AND DEEP LEARNING. In: *Cognex* [online]. [cit. 2022-04-15]. Available at: https://www.cognex.com/what-is/deep-learning/choosing-between-machine-vision-and-deep-learning

[5] SANDHU, Raminder. Machine Vision Promises Nearly Flawless Quality Control. In: *IndustryWeek* [online]. [cit. 2022-04-10]. Available at: https://www.industryweek.com/operations/article/21120473/machine-vision-promises-nearly-flawless-quality-control

[6] Human Vision v Machine Vision (and what the operational benefits really are). In: *Industrial Vision System ltd* [online]. [cit. 2022-04-10]. Available at: https://www.industrialvision.co.uk/news/human-vision-v-machine-vision-and-what-the-operational-benefits-really-are

[7] AI-inspector.one. In: *YouTube* [online]. [cit. 2022-05-08]. Available at: https://www.youtube.com/watch?v=uoFKuZ4bK78

[8] XPlanar | Planar motor system. In: *BECKHOFF* [online]. [cit. 2022-04-13]. Available at: https://www.beckhoff.com/cs-cz/products/motion/xplanar-planar-motor-system/

[9] AI-Inspector.one. In: *BVV* [online]. [cit. 2022-04-13]. Available at: https://www.bvv.cz/en/msv/msv-gold-medal/2021/150-ai-inspectorone/

[10] ZEISS SurfMax: Surface Defect Detection. In: *ZEISS* [online]. [cit. 2022-04-16]. Available at: https://www.zeiss.com/metrology/products/systems/surface-defect-detection.html#benefittwo

[11] Deep Learning Delivers Automated Surface Defect Detection. In: *Metrology.news* [online]. [cit. 2022-04-16]. Available at: https://metrology.news/deep-learning-delivers-automated-surface-defect-detection/

[12] Optical Sorter Mechanical – SOM. In: *Marcelissen* [online]. [cit. 2022-04-24]. Available at: https://marcelissen.com/en/potato-sorting-machine/

[13] Completeness Check. In: *SAC* [online]. [cit. 2022-04-21]. Available at: https://www.sac-vision.de/en/solutions/completeness-check

[14] Application Examples — Steering Wheel Assembly Check. In: *Keyence* [online]. [cit. 2022-04-21]. Available at: https://www.keyence.eu/nlnl/ss/products/vision/iv-casestudy/example/automotive/handle.jsp

[15] PRINTED CIRCUIT BOARD ASSEMBLY VERIFICATION. In: *Cognex* [online]. [cit. 2022-04-22]. Available at: https://www.cognex.com/industries/electronics/pcb-assembly/pcb-assembly-verification

[16] TRIM FINAL ASSEMBLY VERIFICATION. In: *Cognex* [online]. [cit. 2022-04-22]. Available at: https://www.cognex.com/industries/automotive/chassis-systems/trim-final-assembly-verification

[17] Praline Box Inspection. In: *Senswork* [online]. [cit. 2022-04-22]. Available at: https://senswork.com/projects/projects-reader/praline-box-inspection.html

[18] High Accuracy Image Dimension Measurement System. In: *Keyence* [online]. [cit. 2022-04-23]. Available at: https://www.keyence.com/products/measure-sys/image-measure/lm/

[19] 3D Quality Inspection. In: *ABB* [online]. [cit. 2022-04-23]. Available at: https://new.abb.com/products/robotics/application-cells/3d-quality-inspection

[20] Vision-Guided Robots. In: *Keyence* [online]. [cit. 2022-04-24]. Available at: https://www.keyence.com/ss/products/vision/visionbasics/use/inspection07/

[21] Kamerové navádění robotu. In: *Visimatic* [online]. [cit. 2022-04-24]. Available at: https://visimatic.cz/kamerove-navadeni-robotu/

[22] Integrated Vision. In: *ABB* [online]. [cit. 2022-04-24]. Available at: https://new.abb.com/products/robotics/application-equipment-and-accessories/vision-systems/integrated-vision

[23] 3D Vision-Guided Robotics. In: *Keyence* [online]. [cit. 2022-04-24]. Available at: https://www.keyence.com/products/vision/vision-sys/3d_vgr/

[24] Active Pixel Sensor Vs CCD. Who is the clear winner?. In: *Stefano Meroli* [online]. [cit. 2022-04-25]. Available at: https://meroli.web.cern.ch/lecture_cmos_vs_ccd_pixel_sensor.html

[25] Sony's new Leaf Shutter mechanism patent. In: *Sonyalpha Rumours* [online]. [cit. 2022-04-27]. Available at: https://www.sonyalpharumors.com/leaf-shutter-mechanism-patent/

[26] Focal-plane shutter. In: *Wikipedia* [online]. [cit. 2022-04-27]. Available at: https://en.wikipedia.org/wiki/Focal-plane_shutter

[27] Electronic Shutter Types. In: *Basler Product Documentation* [online]. [cit. 2022-04-27]. Available at: https://docs.baslerweb.com/electronic-shutter-types

[28] Rolling Shutter vs Global Shutter sCMOS Camera Mode. In: *Oxford Instruments Andor* [online]. [cit. 2022-04-27]. Available at: https://andor.oxinst.com/learning/view/article/rolling-and-global-shutter

[29] TechnologyPregius/Pregius S. In: *Sony* [online]. [cit. 2022-04-27]. Available at: https://www.sony-semicon.co.jp/e/products/IS/industry/technology.html

[30] PYTHON5000: CMOS Image Sensor, 5.3 MP, Global Shutter. In: *Onsemi* [online]. [cit. 2022-04-27]. Available at: https://www.onsemi.com/products/sensors/image-sensors/python5000

[31] Three-CCD camera. In: *Wikipedia* [online]. [cit. 2022-04-27]. Available at: https://en.wikipedia.org/wiki/Three-CCD_camera

[32] Foveon X3 sensor: What is it and how does it work?. In: *What Digital Camera* [online]. [cit. 2022-04-27]. Available at: https://www.whatdigitalcamera.com/technology_guides/foveon-x3-sensor-what-is-it-and-how-does-it-work-65270

[33] Bayer filter. In: *Wikipedia* [online]. [cit. 2022-04-27]. Available at: https://en.wikipedia.org/wiki/Bayer_filter

[34] AcA2040-35gc - Basler ace. In: *Basler* [online]. [cit. 2022-04-27]. Available at: https://www.baslerweb.com/en/products/cameras/area-scan-cameras/ace/aca2040-35gc/

[35] Go-X Series GOX-3201C-PGE compact 3.2 MP area scan camera. In: *Jai* [online]. [cit. 2022-04-27]. Available at: https://www.jai.com/products/gox-3201c-pge

[36] Blackfly S GigE. In: *Flir* [online]. [cit. 2022-04-27]. Available at: https://www.flir.com/products/blackfly-s-gige/?model=BFS-PGE-50S5C-C&vertical=machine+vision&segment=iis

[37] OBJEKTIV VS TECHNOLOGY APT & VS-TC. In: *W-Technika* [online]. [cit. 2022-04-28]. Available at: https://www.w-technika.cz/objektiv-vs-technology-apt-vs-tc.html

[38] Lens Selector. In: *Basler* [online]. [cit. 2022-04-28]. Available at: https://www.baslerweb.com/en/sales-support/tools/lens-selector/

[39] Understanding Focal Length and Field of View. In: *Edmund Optics* [online]. [cit. 2022-04-28]. Available at: https://www.edmundoptics.co.kr/knowledge-center/application-notes/imaging/understanding-focal-length-and-field-of-view/

[40] Imaging Lenses. In: *Edmund Optics* [online]. [cit. 2022-04-28]. Available at: https://www.edmundoptics.com/c/imaging-lenses/1000/

[41] Why You Should Go Beyond the Kit Lens. In: *BH* [online]. [cit. 2022-04-28]. Available at: https://www.bhphotovideo.com/explora/photography/features/why-you-should-go-beyond-kit-lens

[42] INTRODUCTION TO MACHINE VISION: A guide to automating process quality improvements. In: *Cognex* [online]. 23 [cit. 2022-04-28]. Available at: https://www.cognex.com/resources/white-papers-articles/whitepaperandarticlemain?event=f6c6ef16-20ec-4564-bc74-7c42a9a4900acm_campid=7014W000000ub6yQAA

[43] Machine Vision Lights. In: *Wordop* [online]. [cit. 2022-04-28]. Available at: https://www.wodp.com/products/MachineVisionLights/1.html

[44] First neural network for beginners explained (with code). In: *Towards Data Science* [online]. [cit. 2022-04-30]. Available at: https://towardsdatascience.com/first-neural-network-for-beginners-explained-with-code-4cfd37e06eaf

[45] Underfitting Overfitting—The Thwarts of Machine Learning Models'Accuracy. In: *Towards AI* [online]. [cit. 2022-04-30]. Available at: https://towardsai.net/p/machine-learning/underfitting-overfitting%E2%80%8A-%E2%80%8Athe-thwarts-of-machine-learning-models%E2%80%8Baccuracy

[46] A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way. In: *Towards Data Science* [online]. [cit. 2022-04-30]. Available at: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

[47] Convolutional Neural Network. In: *Towards Data Science* [online]. [cit. 2022-04-30]. Available at: https://towardsdatascience.com/convolutional-neural-network-17fb77e76c05

[48] Pretrained Deep Neural Networks. In: *MathWorks* [online]. [cit. 2022-04-30]. Available at: https://www.mathworks.com/help/deeplearning/ug/pretrained-convolutional-neural-networks.htmlmw_584d069d-c78a-4738-a74b-bf7c5f118472

[49] ImageNet. In: *Image-Net* [online]. [cit. 2022-04-30]. Available at: https://www.image-net.org/

[50] Places. In: *Places* [online]. [cit. 2022-04-30]. Available at: http://places2.csail.mit.edu/download.html

[51] Advanced Guide to Inception v3. In: *Google* [online]. [cit. 2022-04-30]. Available at: https://cloud.google.com/tpu/docs/inception-v3-advanced

[52] TensorFlow. In: *TensorFlow* [online]. [cit. 2022-04-30]. Available at: https://www.tensorflow.org/

[53] Keras. In: *Keras* [online]. [cit. 2022-04-30]. Available at: https://keras.io/

[54] PyTorch. In: *PyTorch* [online]. [cit. 2022-04-30]. Available at: https://pytorch.org/get-started/locally/

[55] Caffe2. In: *Caffe2* [online]. [cit. 2022-04-30]. Available at: https://caffe2.ai/

[56] Deep Learning Toolbox. In: *MathWorks* [online]. [cit. 2022-04-30]. Available at: https://www.mathworks.com/help/deeplearning/index.html?s_tid=CRUX_lftnav

[57] Pypylon. In: *GitHub* [online]. [cit. 2022-05-12]. Available at: https://github.com/basler/pypylon

List of Abbreviations

1D	1-Dimensional
2D	2-Dimensional
2-DOF	2-Degrees Of Freedom
3D	3-Dimensional
A/D	Analog/Digital
Adam	Adaptive Moment Estimation
AFOV	Angular Field of View
AI	Artificial Intelligence
API	Application Programming Interface
CCD	Charge-Coupled Device
CMOS	Complementary Metal–Oxide–Semiconductor
CNN	Convolutional Neural Network
COB	Chip On Board
FOV	Field of View
FPS	Frames Per Second
GPU	Graphics Processing Unit
GUI	Graphical User Interface
I/O	Input/Output
ID	Identification
IDE	Integrated Development Environment
IT	Information Technology
IP	Internet Protocol
JPEG	Joint Photographic Experts Group
LAD	Ladder logic

LED	Light-Emitting Diode
MatLab	Matrix Laboratory
NEMA	National Electrical Manufacturers Association
NN	Neural Network
PC	Personal Computer
PCB	Printed Circuit Board
PLC	Programmable Logic Controller
POE	Power Over Ethernet
PWM	Pulse-Width Modulation
ReLU	Rectified Linear Unit
RGB	Red Green Blue
TCP	Transmission Control Protocol
USB	Universal Serial Bus
VGA	Video Graphics Array
WD	Working Distance

List of Symbols

Symbol	**Name**
H	Horizontal size of the camera sensor
f	Focal length
E_k	Loss function
o_j	Output value
y_kj	Desired value
w_ij	weight
f_y	Ohnisková vzdálenost v y-ovém směru
∇	Gradient
M	Momentum
φ	Angular position
I	Moment of inertia
t	Time

List of Figures

List of Tables

Appendix

Electronical attachment

- 3D model of the optical cell
- Python script for dataset creation
- Python script for dataset editing
- Python script for CNN training
- Python script for testing
- Dampers dataset